



1807.1805

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:)
JACOB ZIMMERMANN, et al.)
Application No.: 09/973,055)
Filed: October 10, 2001)
For: METHOD AND DEVICE FOR)
REMOTELY USING A DATA-)
PROCESSING OBJECT IN A)
COMMUNICATION NETWORK : November 5, 2001

Commissioner for Patents
Washington, D.C. 20231

CLAIM TO PRIORITY

Sir:

Applicants hereby claim priority under the International Convention and all rights to which they are entitled under 35 U.S.C. § 119 based upon the following French Priority Application:

France 0012943, filed October 10, 2000.

A certified copy of the priority document is enclosed.

Applicants' undersigned attorney may be reached in our Costa Mesa, California office by telephone at (714) 540-8700. All correspondence should continue to be directed to our address given below.

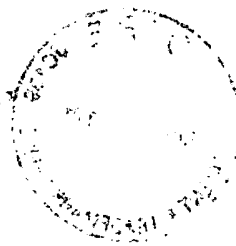
Respectfully submitted,



Attorney for Applicants

Registration No. 42,746

FITZPATRICK, CELLA, HARPER & SCINTO
30 Rockefeller Plaza
New York, New York 10112-3801
Facsimile: (212) 218-2200



THIS PAGE BLANK (USPTO)



BREVET D'INVENTION

CERTIFICAT D'UTILITÉ - CERTIFICAT D'ADDITION

COPIE OFFICIELLE

Le Directeur général de l'Institut national de la propriété industrielle certifie que le document ci-annexé est la copie certifiée conforme d'une demande de titre de propriété industrielle déposée à l'Institut.

Fait à Paris, le **20 SEP. 2001**

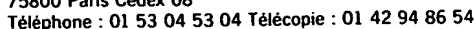
Pour le Directeur général de l'Institut
national de la propriété industrielle
Le Chef du Département des brevets

Martine PLANCHE

INSTITUT
NATIONAL DE
LA PROPRIÉTÉ
INDUSTRIELLE

SIEGE
26 bis, rue de Saint Petersburg
75800 PARIS cedex 08
Téléphone : 33 (1) 53 04 53 04
Télécopie : 33 (1) 42 93 59 30
www.inpi.fr

THIS PAGE BLANK (USPTO)



Code de la propriété intellectuelle - Livre VI



REQUÊTE EN DÉLIVRANCE 1/2

Cet imprimé est à remplir lisiblement à l'encre noire

DB 540 W / 260899

<div style="text-align: center; border: 1px solid black; padding: 2px; font-size: small;">Réservé à l'INPI</div> <div style="display: flex; justify-content: space-between;"> <div> <p>REMISE DES PIÈCES</p> <p>DATE 10 OCT 2000</p> <p>LIEU 75 INPI PARIS</p> <p>N° D'ENREGISTREMENT</p> <p>NATIONAL ATTRIBUÉ PAR L'INPI 0012943</p> <p>DATE DE DÉPÔT ATTRIBUÉE</p> <p>PAR L'INPI 10 OCT. 2000</p> </div> <div> <p>1 NOM ET ADRESSE DU DEMANDEUR OU DU MANDATAIRE À QUI LA CORRESPONDANCE DOIT ÊTRE ADRESSÉE</p> <p>RINUY, SANTARELLI 14, avenue de la Grande Armée 75017 PARIS</p> </div> </div>		
<p>V s références pour ce dossier (facultatif) BIF022767/FR</p>		
<p>C nfirmation d'un dépôt par télécopie <input type="checkbox"/> N° attribué par l'INPI à la télécopie</p>		
<p>2 NATURE DE LA DEMANDE</p>	<p>Cochez l'une des 4 cases suivantes</p>	
<p>Demande de brevet <input checked="" type="checkbox"/></p>		
<p>Demande de certificat d'utilité <input type="checkbox"/></p>		
<p>Demande divisionnaire <input type="checkbox"/></p>		
<p style="text-align: center;"><i>Demande de brevet initiale</i></p>	<p>N° / /</p>	
<p style="text-align: center;"><i>ou demande de certificat d'utilité initiale</i></p>	<p>N° / /</p>	
<p>Transformation d'une demande de brevet européen <i>Demande de brevet initiale</i> <input type="checkbox"/></p>	<p>N° / /</p>	
<p>3 TITRE DE L'INVENTION (200 caractères ou espaces maximum)</p> <p style="text-align: center;">Procédé et dispositif d'utilisation à distance d'un objet informatique dans un réseau de communication</p>		
<p>4 DÉCLARATION DE PRIORITÉ OU REQUÊTE DU BÉNÉFICE DE LA DATE DE DÉPÔT D'UNE DEMANDE ANTÉRIEURE FRANÇAISE</p>	<p>Pays ou organisation / / N°</p> <p>Pays ou organisation / / N°</p> <p>Pays ou organisation / / N°</p> <p><input type="checkbox"/> S'il y a d'autres priorités, cochez la case et utilisez l'imprimé «Suite»</p>	
<p>5 DEMANDEUR</p>	<p><input type="checkbox"/> S'il y a d'autres demandeurs, cochez la case et utilisez l'imprimé «Suite»</p>	
<p>Nom ou dénomination sociale</p>	<p>CANON KABUSHIKI KAISHA</p>	
<p>Prénoms</p>		
<p>Forme juridique</p>	<p>Société de droit Japonais</p>	
<p>N° SIREN</p>		
<p>Code APE-NAF</p>		
<p>Adresse</p>	<p>30-2, Shimomaruko 3-chome, Ohta-ku, Tokyo</p>	
<p>Rue</p>		
<p>Code postal et ville</p>		
<p>Pays</p>	<p>JAPON</p>	
<p>Nationalité</p>	<p>JAPONAISE</p>	
<p>N° de téléphone (facultatif)</p>		
<p>N° de télécopie (facultatif)</p>		
<p>Adresse électronique (facultatif)</p>		

REMISE DES PIÈCES DATE 10 OCT 2000 LIEU 75 INPI PARIS N° D'ENREGISTREMENT 0012943 NATIONAL ATTRIBUÉ PAR L'INPI		Réservé à l'INPI	DB 540 W / 190600
Vos références pour ce dossier : <i>(facultatif)</i>		BIF022767/FR	
6 MANDATAIRE Nom Prénom Cabinet ou Société		RINUY, SANTARELLI	
N° de pouvoir permanent et/ou de lien contractuel			
Adresse	Rue Code postal et ville	14 AVENUE DE LA GRANDE ARMÉE 75017 PARIS	
N° de téléphone <i>(facultatif)</i> N° de télécopie <i>(facultatif)</i> Adresse électronique <i>(facultatif)</i>		01 40 55 43 43	
7 INVENTEUR (S)			
Les inventeurs sont les demandeurs		<input type="checkbox"/> Oui <input checked="" type="checkbox"/> Non Dans ce cas fournir une désignation d'inventeur(s) séparée	
8 RAPPORT DE RECHERCHE		Uniquement pour une demande de brevet (y compris division et transformati n)	
Établissement immédiat ou établissement différé		<input checked="" type="checkbox"/> <input type="checkbox"/>	
Paiement échelonné de la redevance		Paiement en deux versements, uniquement pour les personnes physiqu s <input type="checkbox"/> Oui <input type="checkbox"/> Non	
9 RÉDUCTION DU TAUX DES REDEVANCES		Uniquement pour les personnes physiques <input type="checkbox"/> Requête pour la première fois pour cette invention <i>(joindre un avis de non-imposition)</i> <input type="checkbox"/> Requête antérieurement à ce dépôt <i>(joindre une copie de la décision d'admission pour cette invention ou indiquer sa référence) :</i>	
Si vous avez utilisé l'imprimé «Suite», indiquez le nombre de pages jointes			
10 SIGNATURE DU DEMANDEUR OU DU MANDATAIRE (Nom et qualité du signataire)		VISA DE LA PRÉFECTURE OU DE L'INPI M. MARTIN	
Bruno QUANVIN N°92.1206 RINUY, SANTARELLI			

DÉPARTEMENT DES BREVETS

26 bis, rue de Saint Pétersbourg
75800 Paris Cedex 08

Téléphone : 01 53 04 53 04 Télécopie : 01 42 94 86 54

DÉSIGNATION D'INVENTEUR(S) Page N° .1 / .4.

(Si le demandeur n'est pas l'inventeur ou l'unique inventeur)

Cet imprimé est à remplir lisiblement à l'encre noire

DB 113 W / 260899

V s références pour ce dossier (facultatif)		BIF022767/FR	
N° D'ENREGISTREMENT NATIONAL		00 12943	
TITRE DE L'INVENTION (200 caractères ou espaces maximum)			
Procédé et dispositif d'utilisation à distance d'un objet informatique dans un réseau de communication			
LE(S) DEMANDEUR(S) : CANON KABUSHIKI KAISHA			
DESIGNE(NT) EN TANT QU'INVENTEUR(S) : (Indiquez en haut à droite «Page N° 1/1» S'il y a plus de trois inventeurs, utilisez un formulaire identique et numérotez chaque page en indiquant le nombre total de pages).			
Nom		ZIMMERMANN	
Prénoms		Jacob	
Adresse	Rue	24, rue du 8 Mai 1945	
	Code postal et ville	78460	CHEVREUSE
Société d'appartenance (facultatif)			
Nom		MOREAU	
Prénoms		Jean-Jacques	
Adresse	Rue	91b, rue de Dinan	
	Code postal et ville	35000	RENNES
Société d'appartenance (facultatif)			
Nom			
Prénoms			
Adresse	Rue		
	Code postal et ville		
Société d'appartenance (facultatif)			
DATE ET SIGNATURE(S) DU (DES) DEMANDEUR(S) OU DU MANDATAIRE (Nom et qualité du signataire)		Le 10 octobre 2000 Bruno QUANTIN N°92.1206 RINUY, SANTARELLI	

DOCUMENT COMPORTANT DES MODIFICATIONS

PAGE(S) DE LA DESCRIPTION OU DES REVENDEICATIONS OU PLANCHE(S) DE DESSIN			R.M.*	DATE DE LA CORRESPONDANCE	TAMPON DATEUR DU CORRECTEUR
Modifiée(s)	Supprimée(s)	Ajoutée(s)			
32-38		39-40	oui	15/01/01	18 JAN. 2001 - I F

Un changement apporté à la rédaction des revendications d'origine, sauf si celui-ci découle des dispositions de l'article R.612-36 du code de la Propriété Intellectuelle, est signalé par la mention «R.M.» (revendications modifiées).

5

10 La présente invention concerne un procédé d'utilisation à distance d'un objet informatique accessible via une station serveur connectée à un réseau de communication, à partir d'une station cliente connectée au réseau. Corrélativement, la présente invention concerne un procédé d'exécution d'une fonction sur un objet informatique, mis en œuvre dans une telle station serveur.

15 La présente invention concerne également des dispositifs aptes à mettre en œuvre les procédés précités.

L'invention s'applique de manière générale aux réseaux de communication qui définissent un protocole de communication entre les différents sites reliés au réseau. Un tel réseau peut être à titre d'exemple non
20 limitatif un réseau de communication planétaire tel que l'Internet.

Dans ce qui suit, on entend par "objet informatique" un objet défini selon le fameux concept de programmation désigné par l'expression "programmation orientée objet". Un tel objet informatique est un élément
25 programmatique comprenant des "attributs" ou paramètres (en anglais *arguments*), c.-à-d. des données qui définissent sa structure, et des "méthodes" ou fonctions qui permettent de définir son comportement. De manière classique, les méthodes d'un objet peuvent être appelées (en anglais *invoked*) pour manipuler les données de l'objet.

Le développement d'Internet et du commerce électronique
30 nécessite le déploiement grandissant d'applications de type client-serveur. Dans ces applications, le traitement est effectué par un logiciel tournant sur un serveur accessible depuis le réseau, l'interface homme-machine étant fournie

par un logiciel client s'exécutant sur le poste client et communiquant avec le serveur par l'intermédiaire d'une interface définie.

Le World Wide Web (Web ou WWW) constitue l'interface de prédilection pour accéder à la plupart des ressources disponibles sur Internet.

- 5 Le Web est un ensemble de protocoles Internet et de logiciels qui présentent l'information dans un format hypertexte. Le concept d'hypertexte s'emploie pour construire des documents qui référencent d'autres documents à l'aide de liens, aisément sélectionnables par des utilisateurs novices.

- 10 Parmi les ressources et protocoles développés pour le Web, on trouve l'URL (Uniform Resource Locator), le protocole HTTP (Hypertext Transfer Protocol : protocole de transport hypertexte), le langage HTML (Hypertext Markup Language : langage de marquage hypertexte), le langage XML (Extensible Markup Language : langage de marquage étendu) qui complète le langage HTML, et l'interface CGI (Common Gateway Interface :
15 interface de passerelle commune).

HTTP est un protocole client/serveur qui utilise un modèle requête/réponse. Un client HTTP, ou agent d'utilisateur (souvent un navigateur Web), se connecte à un serveur HTTP en utilisant une URL, et lui adresse une requête relative à une ressource, telle qu'un document HTML.

- 20 Pourtant, jusqu'à ce jour, le Web se montre inadapté à la conception d'applications interactives complexes, qui, restent souvent peu ou pas intégrés au Web, car elles reposent sur l'utilisation de logiciels clients spécifiques. Dans sa forme classique, le Web n'autorise en effet qu'un service de consultation de documents du type requête-réponse, alors qu'un service de
25 traitement de données en ligne (par ex. album photo, compte bancaire etc.) nécessite une infrastructure de manipulation à distance d'objets informatiques.

- Ce défaut est aujourd'hui compensé en partie par l'utilisation de traducteurs et passerelles (par ex. l'interface CGI) qui convertissent les messages transmis entre un serveur et un client, effectuant la transformation
30 d'une représentation utilisable par un navigateur Web client (par ex. des documents HTML véhiculés par le protocole HTTP) vers une représentation intelligible par une application serveur et vice versa. L'utilisation de traducteurs

et passerelles impliquent une complexité de mise en œuvre accrue et une lourdeur excessive de l'interface utilisateur.

La figure 1 représente un exemple de système client-serveur implémenté dans un réseau de communication dans lequel au moins deux applications clientes accèdent à un objet distant hébergé par un serveur, en utilisant deux techniques distinctes connues.

Un logiciel serveur 10 hébergé par une station serveur génère une instance d'un objet 12. Le logiciel serveur 10 est doté par ailleurs de deux interfaces de communication 14, 16 qui permettent de rendre l'objet 12 utilisable sur le réseau par des applications clientes hébergées par des stations clientes (par ex. des micro-ordinateurs), en utilisant des protocoles de communication différents.

Dans cet exemple, l'interface de communication 14 est basée sur une infrastructure connue sous l'acronyme CORBA (*Common Object Request Broker Architecture*, traduisible par "architecture commune de gestionnaire de requêtes à des objets").

L'architecture CORBA développée par le consortium Object Management Group (OMG) permet d'accéder à distance à des objets, quel que soit le langage de programmation utilisé pour leur implémentation. Dans cette architecture, les objets possèdent une description précise de leurs méthodes et des paramètres de celles-ci, exprimée dans le langage spécifique OMG-IDL (*Interface Definition Language*, que l'on peut traduire par "langage de définition d'interfaces") et peuvent communiquer à travers l'Internet par l'intermédiaire du protocole IIOP (*Inter-ORB Protocol*).

Une première application cliente 20 hébergée dans une station cliente utilise les mêmes protocoles que ceux de l'interface CORBA 14 pour pouvoir transmettre ses requêtes à l'objet distant 12, et dans ce but présente à un utilisateur une interface homme-machine (GUI, *graphic user interface*) 22 spécifique.

Les architectures telles que CORBA présentent l'inconvénient de nécessiter une application cliente spécifique implémentant les protocoles et standards définis par l'architecture considérée, et dédiée à cette utilisation.

Cela induit donc un manque important de souplesse d'utilisation pour l'utilisateur. Par ailleurs, tous les protocoles et standards mis en jeu ne sont pas partie intégrante de ceux définis pour le Web.

Toujours à la figure 1, le logiciel serveur 10 est doté d'une
5 seconde interface de communication 16 qui permet de rendre l'objet 12 utilisable sur le réseau par une seconde application cliente 30.

L'interface 16 permet d'encoder l'objet 12 dans un langage de formatage tel que le langage XML pour créer un document XML 32. Ce document XML peut être interprété et visualisé à l'aide d'une application cliente
10 30, telle qu'un navigateur Web.

Bien que, selon cette technique, le logiciel client (navigateur Web) ne soit pas dédié à cette application (utilisation d'objets distants 12), cette technique n'apporte qu'un intérêt limité pour l'utilisateur.

En effet, la représentation en langage de formatage (par ex.
15 langage XML) d'un objet ne permet à l'utilisateur que de visualiser la structure programmatique de l'objet sans aucune possibilité d'utilisation interactive.

Par ailleurs, le brevet américain No 6 078 322 (Simonoff et al.) concerne une technique d'accès à des applications distantes sur un réseau. Plus particulièrement, il est décrit un dispositif client, appelé "client universel" et
20 basé sur une architecture client-serveur, permettant à des systèmes informatiques d'architecture variée reliés à Internet d'exécuter le même logiciel d'application sans modification ou re-compilation de celui-ci.

Ce dispositif utilise un langage de programmation spécifique, le GUIScript, qui peut être exécuté sur un navigateur par le biais d'une application
25 JAVA™ (Applet). Le programme en GUIScript est téléchargé sur le poste client où il s'exécute, construisant une interface utilisateur et assurant la gestion de celle-ci. Dans les cas où le client doit communiquer avec l'application serveur, il lui transmet à son tour du code programme en GUIScript qui sera exécuté par le serveur afin d'y déclencher les opérations désirées.

30 Cependant, bien que permettant à un utilisateur de commander à distance une application logicielle, le dispositif décrit dans le brevet précité ne permet pas l'utilisation à distance d'un objet informatique au sens de la

programmation orientée objet (POO) et tel que défini supra. En outre, ce dispositif requiert l'utilisation d'un langage de programmation spécifique (GUIScript) ne faisant pas partie des outils standardisés du Web, et que doivent supporter à la fois le serveur et le client.

5 Le but de la présente invention est de pallier les insuffisances et défauts de l'art antérieur tels que brièvement exposés ci-dessus. En particulier, la présente invention vise à permettre aux utilisateurs d'un réseau de type Internet de pouvoir accéder à distance à tout type de services en ligne basés sur la mise en œuvre d'objets informatiques, et cela à partir d'une interface
10 client tel qu'un navigateur Web quasiment standard ; l'accès aux services distants impliquant l'utilisation des protocoles et ressources courants à ce jour sur le Web.

A cet effet, la présente invention concerne, selon un premier aspect, un procédé d'utilisation à distance d'un objet informatique accessible via
15 une station serveur connectée à un réseau de communication, à partir d'une station cliente connectée au réseau. Ce procédé est caractérisé en ce qu'il comporte les étapes suivantes :

- génération et envoi d'un message de requête, dit "requête objet", vers la station serveur, ladite requête objet comportant une information
20 d'identification d'un objet accessible via la station serveur ;

- réception d'un message de réponse, dit "réponse objet", émis par la station serveur, ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des fonctions programmées, l'interface permettant à un utilisateur d'utiliser l'objet ;

25 - lancement de l'interface utilisateur sur la station cliente ;
- exécution d'au moins une fonction associée à au moins un élément de l'interface utilisateur, en réponse à l'activation d'au moins un élément par un utilisateur ;

- génération et envoi à la station serveur d'un message de
30 requête, dit "requête d'exécution de méthode", en réponse à l'exécution d'au moins une fonction programmée associée à au moins un élément activé de l'interface utilisateur, ladite requête d'exécution de méthode comportant

l'information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

5 Ainsi, selon l'invention, en faisant une requête classique pour obtenir un objet informatique, une station cliente reçoit en réponse une interface utilisateur de cet objet pouvant être lancée sur la station cliente, et à partir de laquelle un utilisateur de la station cliente peut, à distance, utiliser cet objet comme si celui-ci était hébergé localement dans la station cliente.

10 Selon un mode préféré de réalisation de l'invention, le procédé d'utilisation à distance d'un objet informatique, comporte en outre les étapes suivantes : réception d'un message de réponse, dit "réponse d'exécution de méthode", émis par la station serveur en réponse à la requête d'exécution de méthode, ladite réponse d'exécution de méthode contenant des données indicatives du résultat de l'exécution d'au moins une commande compréhensible par l'objet ; décodage des données contenues dans la réponse
15 d'exécution de méthode et mise à jour si nécessaire de l'interface utilisateur.

De cette manière, l'utilisation de l'objet à distance, par un utilisateur de la station cliente, est interactive avec l'objet, puisque le résultat d'une action de l'utilisateur sur l'objet via son interface utilisateur, est répercuté sur l'interface utilisateur.

20 Selon un deuxième aspect, l'invention concerne un procédé d'exécution d'une fonction sur un objet informatique utilisable, via une station serveur connectée à un réseau de communication, par au moins une station cliente connectée au réseau. Ce procédé est caractérisé en ce qu'il comporte les étapes suivantes, mises en œuvre dans la station serveur :

25 - réception d'un message de requête, dit "requête objet", en provenance d'une station cliente, ladite requête objet comportant une information d'identification d'un objet informatique accessible via la station serveur ;

- génération et envoi à la station cliente d'un message de réponse, dit "réponse objet", ladite réponse objet comportant des informations de
30 description d'une interface utilisateur associées à des fonctions programmées, l'interface utilisateur permettant à un utilisateur d'utiliser ledit objet ;

- réception d'un message de requête, dit "requête d'exécution de méthode", en provenance de la station cliente, ladite requête d'exécution de méthode comportant l'information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

5 Selon un mode préféré de réalisation de l'invention, le procédé d'exécution d'une fonction sur un objet informatique, comporte en outre les étapes suivantes : exécution d'au moins une commande, reçue de la station cliente, sur l'objet informatique ; génération et envoi à la station cliente, d'un message de réponse, dit "réponse d'exécution de méthode", ladite réponse
10 d'exécution de méthode contenant des données indicatives du résultat de l'exécution d'au moins une commande sur l'objet .

 Selon une caractéristique particulière de l'invention, pour être accessible sur le réseau, l'objet informatique est associé dans la station serveur à un document électronique contenant les informations de description d'au
15 moins une interface utilisateur et les fonctions programmées associées. Par ailleurs, selon cette caractéristique, l'information d'identification de l'objet est une adresse électronique associée audit document électronique.

 De cette façon, l'objet est rendu accessible sur le réseau par la publication de son interface utilisateur (le document électronique) qui pourra
20 ainsi être téléchargée dans une station cliente.

 Selon un mode de réalisation préféré de l'invention, dans lequel le réseau est l'Internet, la station cliente et la station serveur communiquent en utilisant le protocole HTTP. Dans ce mode de réalisation, l'interface utilisateur de l'objet distant est fournie à la station cliente sous la forme d'un document
25 électronique, structuré selon un langage de marquage tel que le langage XML, et contenant des informations descriptives de l'interface, associées à des fonctions programmées. Ce document électronique associé à l'objet est alors identifié par une adresse électronique de type URL. Les fonctions programmées sont implémentées, par exemple, dans le langage Javascript.

30 Ainsi, par le choix de langages et protocoles standards du Web, la présente invention peut être implémentée dans tout type de terminal Internet

utilisateur courant, tel que par exemple : ordinateur personnel, "set-top box", borne Internet, téléphone mobile, etc.

En particulier, le dispositif d'utilisation à distance d'un objet informatique, selon l'invention, pourra être incorporé dans un navigateur Internet (navigateur Web) en impliquant une surcharge minimale en termes de puissance de calcul et de capacité mémoire requise.

Corrélativement, l'invention concerne un dispositif d'utilisation à distance d'un objet informatique, ainsi qu'un dispositif d'exécution d'une fonction sur un objet informatique, caractérisés en qu'ils comportent respectivement, des moyens adaptés à la mise en œuvre d'un procédé d'utilisation à distance d'un objet informatique, tel que défini supra, et des moyens adaptés à la mise en œuvre d'un procédé d'exécution d'une fonction sur un objet informatique, tel que défini supra.

L'invention vise également une station cliente reliée à un réseau de communication, caractérisée en ce qu'elle comporte un dispositif d'utilisation à distance d'un objet informatique, tel que défini supra.

L'invention vise encore une station serveur reliée à un réseau de communication, caractérisée en ce qu'elle comporte un dispositif d'exécution d'une fonction sur un objet informatique, tel que défini supra.

L'invention vise aussi un réseau de communication comportant au moins une station cliente en conformité avec l'invention, et au moins une station serveur en conformité avec l'invention.

L'invention vise également un système informatique comportant des moyens adaptés à mettre en œuvre au moins un des procédés selon l'invention tels qu'exposés supra.

L'invention vise aussi un programme d'ordinateur comportant une ou plusieurs séquence d'instructions apte à mettre en œuvre au moins un des procédés selon l'invention tels qu'exposés supra, lorsque ce programme est exécuté par un ordinateur.

L'invention vise encore un support d'informations, tel qu'une disquette ou un compact disque (CD), caractérisé en ce qu'il contient un tel programme d'ordinateur.

Les avantages de ces dispositifs, stations, de ce système informatique, de ce programme d'ordinateur, et de ce support d'informations sont identiques à ceux des procédés tels que succinctement exposés ci-dessus.

5 D'autres particularités et avantages de l'invention apparaîtront encore dans la description ci-après. Aux dessins annexés donnés à titre d'exemples non limitatifs :

- la figure 1 représente un exemple de système client-serveur implémenté dans un réseau de communication, dans lequel plusieurs applications clientes accèdent à un objet distant hébergé par un serveur, en utilisant des techniques connues ;

- la figure 2 illustre schématiquement des réseaux de communication adaptés à mettre en œuvre la présente invention ;

- la figure 3 est un schéma fonctionnel illustrant les éléments d'un dispositif selon l'invention, d'exécution d'une méthode associée à un objet informatique, et qui sont incorporés dans une station serveur ;

- la figure 4 est un schéma fonctionnel illustrant les éléments d'un dispositif selon l'invention, d'utilisation à distance d'un objet, et qui sont incorporés dans une station cliente ;

20 - les figures 5 et 6 sont des organigrammes détaillant le procédé selon l'invention d'utilisation à distance d'un objet informatique, mis en œuvre dans une station cliente, et plus particulièrement, la figure 5 détaille le processus de connexion d'une station cliente à un objet distant, tandis que la figure 6 détaille le processus de traitement d'une action d'un utilisateur sur un élément de l'interface utilisateur associé à l'objet distant ;

- la figure 7 est un organigramme détaillant le procédé selon l'invention d'exécution d'une méthode associée à un objet informatique, mis en œuvre dans une station serveur ;

30 - la figure 8 représente schématiquement un ordinateur adapté à mettre en œuvre un procédé d'utilisation à distance d'un objet informatique, selon l'invention, et/ou un procédé d'exécution d'une méthode associée à un objet informatique, selon l'invention.

La présente invention s'applique en général aux réseaux de communication informatique.

Elle s'applique particulièrement bien aux réseaux de communication de grande étendue dans lesquels un très grand nombre de serveurs
5 informatiques sont reliés entre eux.

Un tel réseau peut être à titre d'exemple un réseau de communication planétaire, tel que le réseau Internet, bâti au-dessus d'un protocole de communication qui permet aux ordinateurs connectés au réseau d'échanger des documents.

10 Un tel réseau est illustré par exemple à la figure 2.

Dans cet exemple, et à titre non limitatif, trois réseaux informatiques R1, R2 et R3 sont des réseaux du type Ethernet, connectés entre eux par exemple par l'intermédiaire du réseau Internet (1).

Chaque réseau R1, R2 et R3 comporte un ou plusieurs ordinateurs.
15 Ici, à titre d'exemple, le réseau R1 comporte deux ordinateurs C1, C2 ; le réseau R2 comporte un seul ordinateur C3 ; et le réseau R3 comporte trois ordinateurs C4, C5 et C6.

Chacun de ces ordinateurs C1, C2, C3, C4, C5, C6 est donc capable d'envoyer et de recevoir des données de l'un des quelconques autres
20 ordinateurs.

Dans un tel réseau, il est courant qu'un système d'information, où l'information est stockée dans des documents, soit bâti au-dessus du réseau de communication.

Un tel système d'information peut être un système hypertexte, de sorte que les documents stockés comportent des liens hypertextes, appelés
25 également pointeurs, reliant certains documents entre eux. En d'autres termes, des documents comportent des pointeurs pointant vers d'autres documents, et un utilisateur du réseau peut demander la transmission de ces autres documents en actionnant les pointeurs d'un document initial.

30 A titre d'exemple, le système hypertexte classique bâti au-dessus du réseau de communication Internet est le système WWW, le protocole de

communication utilisé pouvant être le protocole de transfert hypertexte HTTP (*Hyper Text Transfer Protocol*).

5 Dans un tel réseau de communication, l'ensemble des documents regroupés dans la mémoire d'un serveur informatique forme un site informatique, de sorte que le réseau de communication permet de relier un très grand nombre de sites entre eux. On comprendra aisément que chacun des ordinateurs C1 à C6 illustrés à la figure 2 peut être tour à tour un serveur informatique adapté à servir des documents en réponse à des requêtes émises sur le réseau, ou un utilisateur du réseau de communication (appelé encore
10 client), adapté à émettre des requêtes pour demander des documents sur ce réseau.

Outre ces documents formant un système d'information classique sur un réseau de communication, les ordinateurs C1 à C6 peuvent également stocker et créer des objets informatiques grâce à des applications
15 informatiques.

Comme exposé supra, on définit un objet informatique comme un élément comprenant à la fois différents attributs (paramètres), et un ensemble de fonctions (méthodes) qui permettent de manipuler les attributs de l'objet et d'en modifier la valeur. Cet ensemble de fonctions et d'attributs forme une
20 interface de l'objet informatique.

Chaque objet informatique peut être créé dans un langage de programmation utilisé par une application informatique. De tels langages de programmation sont connus par exemple sous le nom de JAVA ou C++.

Il est classique que dans un tel système d'objets distribués, ces
25 objets informatiques pointent les uns vers les autres, c'est-à-dire que certains champs de données d'un objet sont eux-mêmes des objets informatiques.

En liaison avec la **figure 3**, on va tout d'abord décrire un dispositif d'exécution d'une fonction associée à un objet informatique, en conformité avec l'invention. Ce dispositif, appelé ci-après "dispositif-serveur", est implémenté
30 dans une station serveur reliée à un réseau de communication telle que par exemple l'ordinateur C3 dans le réseau de la figure 1.

Comme représenté à la figure 3, le dispositif-serveur 300 comporte une base 307 d'objets publiés. Cette base est constituée d'objets informatiques rendus accessibles (publication) à l'ensemble des stations clientes du réseau 1.

5 Dans cette base d'objets 307, conformément à l'invention, pour être publié, chaque objet est associé à un document électronique 3071 contenant des informations de description d'une interface utilisateur permettant à un utilisateur d'utiliser l'objet. Ces informations de description sont associées dans le document électronique 3071 à des fonctions programmées sensibles à des actions de l'utilisateur sur l'interface utilisateur de l'objet.

10 En pratique les documents 3071 sont des fichiers contenant une description programmée d'une interface graphique (GUI, graphic user interface) permettant de mettre en œuvre l'objet considéré.

Dans un mode de réalisation préféré, les documents 3071 sont des documents XML, c.-à-d., dans lesquels l'interface graphique de l'objet est
15 décrite au moyens du langage d'annotations XML (*extended markup language*).

En pratique, dans ce mode de réalisation, on utilise la syntaxe du langage XUL (*XML-based User Interface Language*) destiné à la construction d'interface utilisateurs, et développé par l'Organization Mozilla à partir du langage XML, dans le cadre du développement de son navigateur Web :
20 "Mozilla". Pour obtenir plus d'informations sur le langage XUL, on peut consulter le site Web : <http://www.mozilla.org/> .

Chaque document 3071 (désigné par "document XUL"), créé préalablement à la publication, par exemple manuellement, décrit les éléments dont l'interface utilisateur de l'objet est composée, tels que des boutons, des
25 menus ou des champs de saisie.

Conformément à l'invention, dans chaque document XUL, aux instructions XUL sont associées des fonctions programmées implémentant le traitement nécessaire à l'activation de chaque élément (graphique) de l'interface utilisateur de l'objet considéré. Selon un mode de réalisation préféré, ces
30 fonctions sont implémentées dans le langage de script bien connu : JavaScript.

Toujours selon l'invention, on associe à chaque document XUL, une adresse électronique par laquelle une station cliente peut accéder audit

document. En pratique, cette adresse électronique est une adresse URL (*uniform resource locator*). Cette adresse électronique est par conséquent indicative de l'emplacement de stockage du document XUL auquel elle est associée, et par extension, elle est indicative de l'emplacement de stockage de l'objet informatique auquel est associé le document XUL considéré, puisque ce dernier représente l'objet dans la base d'objets publiés 307.

De cette manière, l'adresse électronique (URL) associée à un document XUL 3071 constitue une information d'identification de l'objet correspondant au document XUL considéré.

De préférence, à chaque objet publié est associé un second document électronique 3072 dans lequel est décrit l'interface programmatique de l'objet (c.-à-d. les méthodes et attributs de l'objet). Ce document utilise le langage XML pour représenter l'interface programmatique d'un objet informatique. Une adresse électronique (URL) est également attribuée au document XML ainsi obtenu, de manière à rendre accessible (publication) sur le réseau l'interface programmatique de l'objet (méthodes et attributs).

La traduction du langage de programmation orientée objet – par ex. le langage C++ – dans le langage de communication utilisé, ici le langage XML, utilise une unité de traduction 305.

Cette unité de traduction 305 (désigné par *marshaller* en anglais) permet de traduire un objet informatique créé dans un langage de programmation (par ex. langage C++), vers un langage de communication tel que le langage XML utilisable dans le protocole de communication HTTP du réseau Internet.

On pourra obtenir plus d'informations sur les modalités de cette traduction, ainsi que sur celles de la traduction inverse (XML vers C++) en consultant la demande de brevet européen N° EP 00401754.7 déposée au nom de la société Canon Research Centre France S. A.

En **Annexe I**, on donne un exemple d'une interface programmatique d'un objet informatique ("document XML"). L'objet informatique donné à titre d'exemple est un "processeur d'images" et a pour nom "*ImageProcessor*". Cet objet est publié sur un serveur ayant pour nom

"tournesol" et comporte un certain nombre de fonctions ou méthodes :
 "buyImage", "negative", "convertToBW", "getImage".

Chacune des ces fonctions comporte un certain nombre d'attributs ou paramètres (arguments). Par exemple, la fonction "buyImage" retourne un
 5 résultat de type entier (*int*) et comporte un seul attribut ayant pour nom "name" et de type chaîne de caractères (*string*).

En Annexe II, on donne un exemple de document XUL définissant une interface utilisateur de type graphique permettant à un utilisateur d'utiliser à distance l'objet "ImageProcessor".

10 Dans cet exemple, les fonctions Javascript associées aux éléments graphiques de l'interface, sont listées en premier. Ainsi, un certain nombre de fonctions sont déclarées, parmi lesquelles la fonction "buyNewImage". Comme on peut le voir, la fonction *buyNewImage* permet d'appeler (on dit aussi : invoquer) la méthode *buyImage* de l'objet
 15 *ImageProcessor*.

Toujours en liaison avec l'Annexe II, après la déclaration des fonctions Javascript, on trouve dans le document XUL, des déclarations de variables et des instructions Javascript destinées à initialiser l'affichage de l'interface utilisateur.

20 Ces instructions d'initialisation consiste notamment à fixer des champs de saisie, à préparer des listes de sélection, présélectionner des options etc. Ces instructions peuvent avoir également pour but d'établir une connexion avec un autre serveur distant et préparer le dialogue avec d'autres objets, par exemple en s'enregistrant auprès d'une banque électronique et en
 25 fournissant au serveur associé des certificats nécessaires au fonctionnement d'un protocole de paiement électronique tel que celui connu sous la désignation "Payword".

Ainsi dans l'Annexe II, la fonction "connectToServers" est invoquée pour établir dans cet exemple une connexion avec le serveur
 30 d'images "tournesol". La fonction *connectToServers* est déclarée dans les fonctions Javascript.

Le document XUL donné en exemple comporte, après les instructions d'initialisation de l'interface utilisateur graphique (*graphic user interface*, GUI) associée à l'objet *ImageProcessor*, les instructions XUL proprement dites, destinées à décrire les éléments graphiques de l'interface utilisateur de l'objet. Ainsi, dans cet exemple, un bouton (case de commande) portant la mention "*Buy New*" affiché à l'écran est associé à la fonction Javascript "*buyNewImage*" destinée à soumettre au serveur une requête d'achat d'une nouvelle image.

De retour à la **figure 3**, conformément à l'invention, le dispositif-serveur 300 comporte des moyens pour recevoir un message de requête, désigné ici par "requête objet", en provenance d'une station cliente ; cette requête objet comportant une information d'identification d'un objet informatique accessible via la station serveur. Cette information d'identification est constituée en pratique de l'URL associée au document XUL correspondant à l'objet considéré dans la base d'objets publiés 307.

Ces moyens de réception sont incorporés dans un module appelé communément "serveur HTTP" 302 et assurant le service de communication avec des stations clientes selon le protocole HTTP.

Le dispositif-serveur 300 comporte aussi des moyens pour générer et envoyer à une station cliente un message de réponse, dit "réponse objet", comportant les informations de description d'un interface utilisateur d'un objet informatique demandé par la station cliente, ces informations étant associées à des fonctions programmées. Comme exposé précédemment, ces informations de description d'un interface utilisateur, sont en pratique contenues dans un document XUL tel que décrit plus haut en liaison avec l'Annexe II.

Les moyens précités de génération et d'envoi d'une réponse objet sont incorporés, d'une part dans le serveur HTTP 302, et d'autre part, dans une unité 304 d'identification d'objets et d'extraction de méthodes. Le serveur HTTP assure l'envoi de la requête, alors que l'unité 304 génère le contenu du message HTTP (corps du message).

Le dispositif-serveur 300 comporte encore des moyens pour recevoir un message de requête, désigné ici par "requête d'exécution de

méthode", en provenance d'une station cliente, et comportant l'URL pointant sur le document XUL associé à un objet dans la base d'objets publiés 307. Comme cela sera exposé plus bas, la requête d'exécution de méthode comporte au moins une commande compréhensible par l'objet utilisé par la station cliente.

5 En pratique, une "commande compréhensible" consiste en la désignation d'une méthode de l'objet considéré et en la délivrance de valeurs données aux attributs (arguments) de la méthode désignée.

Les moyens précités de réception d'une requête d'exécution de méthode sont également incorporés dans le serveur HTTP 302.

10 Conformément à l'invention, le dispositif-serveur 300 comporte en outre des moyens pour exécuter une commande incorporée dans une requête d'exécution de méthode, reçue de la station cliente, et portant sur un objet informatique. Ces moyens sont répartis dans l'unité 304 d'identification d'objets et d'extraction de méthodes, et dans l'unité de traduction 305.

15 Pour exécuter une fonction particulière sur un objet informatique, il faut pouvoir disposer de cet objet sous forme de code programme orienté objet, ici en code C⁺⁺. C'est ainsi que le dispositif-serveur 300 comporte un serveur d'objets C⁺⁺, ainsi qu'au moins une application logicielle 309 permettant de mettre en œuvre ces objets.

20 Finalement, le dispositif-serveur 300 comporte des moyens pour générer et envoyer, à une station cliente, un message de réponse, désigné ici par "réponse d'exécution de méthode", et contenant des données indicatives du résultat de l'exécution d'une ou plusieurs commandes sur l'objet informatique utilisé. Ces moyens sont fournis par, d'une part, l'unité de traduction 305 et

25 l'unité 304 (moyens de génération de la réponse), et d'autre part, par le serveur HTTP 302 (moyens d'envoi de la réponse).

En liaison avec la figure 4, on va maintenant décrire un dispositif d'utilisation à distance d'un objet informatique, en conformité avec l'invention. Ce dispositif, appelé ci-après "dispositif-client", est implémenté dans une station

30 cliente reliée à un réseau de communication telle que par exemple les ordinateurs C1, C2, C4, C5 ou C6 dans le réseau de la figure 1.

Le dispositif-client 400, incorporé dans une station cliente reliée au réseau 1, comporte des moyens pour communiquer avec une station serveur en utilisant un protocole de communication tel que le protocole HTTP. Ces moyens de communication sont fournis par un module appelé communément

5 "client HTTP" 402.

Conformément à l'invention, le dispositif-client comporte des moyens pour générer et envoyer un message de requête, désigné ici par "requête objet", vers une station serveur, ladite requête objet comportant une information d'identification d'un objet choisi par un utilisateur et accessible via la station

10 serveur. Dans un mode de réalisation préféré, cette information d'identification est constituée de l'URL associée au document XUL correspondant à l'objet informatique choisi.

Ces moyens de génération et d'envoi d'une requête objet sont incorporés dans une unité 406 dite "de génération de requêtes objets" (génération), et dans le client HTTP 402 (envoi).

15

De même, conformément à l'invention, le dispositif-client 400 comporte des moyens pour recevoir un message de réponse, désigné ici par "réponse objet", émis par la station serveur, ladite réponse objet comportant des informations de description d'une interface utilisateur, associées à des

20 fonctions programmées (document XUL). Cette interface utilisateur permet à un utilisateur d'utiliser, c.-à-d. manipuler, commander, à distance l'objet choisi. Ces moyens de réception sont fournis par le client HTTP 402.

Conformément à l'invention, le dispositif-client 400 comporte également des moyens pour lancer l'interface utilisateur de l'objet sur la station

25 cliente. Ces moyens sont fournis, d'une part, par une unité 404 de traitement XML/XUL permettant de décoder des documents utilisant la syntaxe propre au langage XML et plus particulièrement au langage XUL ; et une unité de traitement Javascript 410 qui coopère avec l'unité 404 pour interpréter les fonctions Javascript incorporées dans un document XUL tel que décrit supra.

30 D'autre part, une unité 408 dite "de présentation d'interface utilisateur d'objets" est chargée de présenter à l'utilisateur l'interface de l'objet.

Ainsi, après traitement (c.-à-d. décodage) d'un document XUL par l'unité 404 en coopération avec l'unité 410, l'interface utilisateur de l'objet sélectionné est lancée par l'unité 408 et présentée à l'utilisateur.

En pratique, dans un mode de réalisation préféré de l'invention, l'interface utilisateur de l'objet est une interface graphique (GUI) présentée à l'utilisateur sous la forme d'une fenêtre de dialogue sur un écran d'affichage (non représenté). L'utilisateur pourra alors utiliser un clavier ou un dispositif de pointage tel qu'une souris pour dialoguer avec l'interface graphique de l'objet.

Conformément à l'invention, le dispositif-client 400, comporte des moyens pour exécuter au moins une fonction programmée associée à un élément de l'interface graphique, en réponse à l'activation de l'élément par l'utilisateur. Ces moyens sont fournis par l'unité 410 de traitement Javascript qui coopère avec l'unité de présentation 408, pour exécuter les fonctions Javascript associées aux éléments graphiques de l'interface utilisateur de l'objet, lorsque ceux-ci sont activés par l'utilisateur.

Le dispositif-client comporte encore des moyens pour générer et envoyer à la station serveur un message de requête, désigné ici par "requête d'exécution de méthode", et comportant une information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

Comme mentionné plus haut, une "commande compréhensible" consiste en la désignation d'une méthode de l'objet considéré et en la délivrance de valeurs données aux attributs (arguments) de la méthode désignée, en réponse à l'activation d'un élément graphique de l'interface utilisateur (par ex. bouton, ascenseur) par l'utilisateur.

Dans le mode de réalisation décrit ici, la commande destinée à l'objet distant est générée par l'unité de traitement Javascript 410 suite à l'exécution d'une fonction Javascript associée à l'interface utilisateur de l'objet. Cette "commande" est ensuite fournie par l'unité 410, à l'unité de traitement XML 404 pour être encodée en langage XML. La commande est ensuite délivrée à l'unité 406 de génération de requêtes objets. L'unité 406 est alors chargée de générer le message de requête, tandis que le module client HTTP 402 assure l'envoi de la requête via le réseau 1 à la station serveur.

Conformément à l'invention, le dispositif-client 400 comporte en outre, des moyens pour recevoir, en réponse à la requête d'exécution de méthode envoyée préalablement, un message de réponse, désigné ici par "réponse d'exécution de méthode", émis par la station serveur. La réponse d'exécution de méthode contient des données indicatives du résultat de l'exécution d'une ou plusieurs commandes contenues dans la requête d'exécution de méthode sur l'objet informatique considéré. Ces moyens de réception sont fournis par le client HTTP 402.

Le dispositif-client comporte également des moyens pour décoder les données contenues dans une telle réponse d'exécution de méthode. Ces moyens sont fournis par l'unité de traitement XML / XUL 404.

Le dispositif-client 400 comporte finalement des moyens pour mettre à jour si nécessaire l'interface utilisateur, suite à la réception d'une réponse d'exécution de méthode. Ces moyens sont fournis par l'unité de traitement Javascript 410 et l'unité 408 de présentation d'interface utilisateur d'objets.

On notera ici que le dispositif-client 400 peut être incorporé dans son ensemble dans un même composant logiciel. En particulier, le dispositif-client pourra être incorporé dans un navigateur Internet (c.-à-d. navigateur Web).

En référence aux figure 5 et 6, on va maintenant décrire le procédé, selon l'invention, d'utilisation à distance d'un objet informatique, mis en œuvre dans une station cliente. Plus particulièrement, en liaison avec la figure 5 on va détailler le processus de connexion d'une station cliente à un objet distant. Puis, en liaison avec la figure 6, on détaillera le processus de traitement d'une action d'un utilisateur sur un élément de l'interface utilisateur associé à un objet distant .

En liaison avec la **figure 5**, un procédé d'utilisation à distance d'un objet informatique, en conformité avec l'invention, débute par une étape E500 d'envoi d'une "requête objet" telle que définie plus haut, vers la station serveur. La requête objet comporte une information d'identification d'un objet informatique accessible via la station serveur. Cette information d'identification

est, dans le mode de réalisation décrit, L'URL pointant sur le document XUL, décrivant l'interface utilisateur de l'objet, publié dans la station serveur.

En pratique, lorsque le dispositif-client 400 est un navigateur Web, l'utilisateur entre, par exemple au clavier, l'URL correspondant à l'objet qu'il
5 désire utiliser.

Par exemple, si l'on considère l'objet "ImageProcessor" donné en Annexe, cette URL pourra être la suivante :

<http://tournesol:8080/web-obj/imageprocessor.xul>

La requête objet envoyée par la station cliente, est un message de
10 requête HTTP utilisant la méthode "GET" du protocole HTTP. La commande HTTP "GET" est utilisée classiquement pour récupérer une ressource d'un serveur. Dans le cadre de l'invention, on utilise la commande GET pour récupérer l'interface utilisateur d'un objet distant, de la même manière que pour obtenir n'importe quelle page Web ordinaire.

15 A l'étape suivante (E502), la station-cliente reçoit un message de réponse (HTTP) de la station serveur. Ce message de réponse contient un document XUL, décrivant l'interface utilisateur de l'objet, tel que celui fourni en Annexe II.

Le document XUL est alors décodé, conjointement par l'unité de
20 traitement XML/XUL 404 et l'unité de traitement Javascript 410. Puis le lancement de l'interface graphique sur la station cliente est réalisé par l'unité de présentation 408 en coopération avec l'unité de traitement Javascript 410. Le lancement de l'interface utilisateur s'opère comme suit.

Dans un premier temps (étape E504), l'interface utilisateur est
25 construit c.-à-d. affiché à l'écran. Puis, dans un deuxième temps (étape E506) l'interface utilisateur est initialisé.

Ensuite (étape 508), le dispositif-client se met dans un état d'attente des actions de l'utilisateur sur l'interface utilisateur. On peut donc dire, qu'à cet instant, l'utilisateur est connecté à l'objet distant.

30 La suite du procédé, selon l'invention, d'utilisation à distance d'un objet informatique, va maintenant être décrite en liaison avec la figure 6.

Dans la **figure 6**, à l'étape E600, l'utilisateur opère une action sur l'interface utilisateur de l'objet distant. Une action de l'utilisateur correspond à l'activation d'au moins un élément de l'interface utilisateur, par ex. un élément graphique tel qu'un bouton.

5 En réponse à une action de l'utilisateur, une fonction programmée associée à l'élément activé est exécutée. Dans le mode de réalisation préféré décrit ici, cette fonction programmée est une fonction Javascript telle que celles données en exemple à l'Annexe II. Ainsi, si l'utilisateur active à l'écran le bouton "Buy New" à l'aide d'une souris, la fonction *buyNewImage* est exécutée.

10 L'étape suivante E604 est une étape de test dans laquelle il est déterminé si la fonction Javascript appelée, induit un appel de méthode d'un objet distant. En effet, il se peut que la fonction exécutée en réponse à une action de l'utilisateur ne concerne pas directement l'objet distant, par exemple, pour modifier un paramètre d'affichage de l'interface graphique.

15 Si c'est le cas (E606), la fonction est exécutée localement dans la station cliente, par le dispositif-client (par ex. un navigateur Web). Le dispositif-client se replace alors dans un état d'attente de la prochaine action de l'utilisateur (E600).

20 Au contraire, si la fonction Javascript exécutée génère un appel de méthode de l'objet distant, le dispositif-client génère et envoie à la station serveur un message de requête – la requête d'exécution de méthode mentionnée supra – comportant l'URL du document XUL correspondant à l'objet, et un appel de méthode de l'objet.

25 A cet effet, dans une première étape, E608, il est procédé à l'encodage dans le langage XML de l'appel de la méthode. A cet effet, la désignation de la méthode appelée et les paramètres associés sont fournis par l'unité de traitement Javascript 410 à l'unité de traitement XML/XUL 404.

30 A titre d'exemple, dans l'Annexe II, si l'appel de méthode est généré par la fonction Javascript "buyNewImage", la méthode appelée est "buyImage" et le paramètre associé est contenu par la variable "name". Dans cet exemple, on suppose que la variable "name" a pour valeur "photo1.bmp" entrée par l'utilisateur au travers de l'interface graphique de l'objet distant.

L'Annexe III donne un exemple d'appel de méthode encodé selon l'invention. On y reconnaît la désignation de la méthode ("function name") invoquée "buyImage" et la valeur des paramètres ("arguments") de cette méthode : "photo1.bmp". Dans cet exemple, l'utilisateur désire acheter l'image numérique identifiée par "photo1" dans le serveur d'images auquel est associé l'objet informatique distant "ImageProcessor" (Annexe I).

Un fois l'encodage XML de l'appel de méthode effectué, la requête d'exécution de méthode contenant l'appel de méthode précité est générée par l'unité 406 de génération de requête objets (étape E610). Puis la requête est envoyée à la station serveur par le module client HTTP 402 (étape E612).

La requête d'exécution de méthode envoyée par la station cliente, est un message de requête HTTP utilisant la méthode "POST" du protocole HTTP. La commande HTTP "POST" est utilisée classiquement pour envoyer des données d'un client à un serveur.

Toujours à la figure 6, à l'étape suivante, E614, le dispositif-client se met dans un état d'attente d'une réponse de la station serveur, désignée par "réponse d'exécution de méthode", cette réponse contenant des données indicatives du résultat de l'exécution de la méthode invoquée de l'objet.

Lorsque la station cliente reçoit (E614) un message de réponse (réponse d'exécution de méthode) en provenance de la station serveur, celui-ci est analysé (étape E616) par le module client HTTP 402, afin de déterminer si l'exécution distante de la méthode invoquée de l'objet distant, a donné lieu à une erreur ou à un résultat valide.

Si le résultat de l'exécution contenu dans le message de réponse est un message d'erreur, l'unité de traitement XML/XUL (404) associée à l'unité de traitement Javascript (410) génère l'exception correspondante (étape E618), c.-à-d. assure le traitement prévu pour ce type d'anomalie d'exécution. Dans ce cas, le processus d'utilisation à distance est interrompu.

Si le résultat est valide, il est procédé au décodage du résultat (étape E620). Ce décodage, réalisé par l'unité de traitement XML/XUL (404) associée à l'unité de traitement Javascript (410), consiste notamment à traduire le résultat reçu, du langage XML vers un format dans lequel ce résultat peut

être exploité par l'interface utilisateur de l'objet distant. Dans le cadre du mode de réalisation décrit ici, le résultat est traduit en Javascript de manière à pouvoir être exploitable par le code programme Javascript appelant (c.-à-d. le code de la fonction Javascript appelante).

- 5 Une fois le résultat décodé, l'interface utilisateur de l'objet est mise à jour (E622) avec le résultat, de manière à rendre compte à l'utilisateur du résultat de sa requête.

10 Finalement, le dispositif utilisateur se replace dans un état d'attente d'une autre action de l'utilisateur sur l'interface graphique de l'objet (étape E600).

L'Annexe IV donne un exemple du contenu d'une réponse d'exécution de méthode générée par la station serveur et envoyée ensuite à la station cliente.

15 Dans cet exemple, il s'agit du résultat de l'exécution de la méthode "buyImage" (cf. Annexe III) associée à l'objet distant "ImageProcessor" (cf. Annexe I), et invoquée par l'exécution de la fonction Javascript "BuyNewImage" (cf. Annexe II). Le résultat est encodé en langage XML. Le marqueur <wob> signifie "web object", le résultat consiste en la valeur entière "10". Dans le cadre de cet exemple, cette valeur "10" est un numéro de série correspondant à
20 l'image numérique "achetée" par l'utilisateur. Ce numéro de série identifie l'image "achetée" de manière unique et permet ensuite à l'utilisateur d'accéder librement à cette image, à la seule condition de fournir ce numéro de série.

25 Dans cet exemple, la valeur "10" est décodée et fournie ensuite à l'unité de traitement Javascript 410 qui la transmet au code de la fonction BuyNewImage appelante. Cette dernière poursuit son exécution et, par les instructions qui suivent celle à l'origine de l'appel de méthode distante, permet la mise à jour de l'interface utilisateur avec la valeur retournée ("10") par la méthode distante ("buyImage").

30 En référence maintenant à la **figure 7**, on va décrire le procédé selon l'invention d'exécution d'une méthode associée à un objet informatique, mis en œuvre dans la station serveur.

A la figure 7, à l'étape E700, le serveur HTTP 302 de la station serveur, reçoit une requête HTTP en provenance d'une station cliente. A l'étape suivante (E702), le serveur HTTP procède à l'extraction de l'URL contenue dans la requête.

5 A l'étape suivante (E704), l'unité d'identification 304 détermine si l'URL extraite pointe sur un objet informatique ou non. Dans la négative – par ex. l'URL pointe sur une simple page web – un traitement approprié est exécuté (E706).

Dans l'affirmative, on détermine si la requête reçue est un requête
10 HTTP POST (étape E708).

Si la requête n'est pas une requête POST, il peut s'agir alors d'une requête GET, par exemple, destinée à obtenir un document XUL associé à un objet publié dans la base 307. Dans ce cas, on applique le traitement approprié (étape E706), par ex. envoi du document XUL demandé.

15 Inversement, si la requête reçue est une requête POST, cette condition ajoutée au fait que l'URL contenue dans la requête pointe sur un objet (E704), permet de déterminer qu'il s'agit d'une requête d'exécution de méthode.

Dans ce cas, l'unité 304 identifie la méthode (étape E710) (par ex. la méthode "buyImage"). Ensuite, à l'étape suivante (E712), la valeur des
20 paramètres associés à la méthode identifiée sont décodés (par ex. "photo1.bmp").

A l'étape qui suit (E714), on procède au décodage de la méthode, c.-à-d. à la traduction de la méthode identifiée avec ses arguments, du langage XML vers le langage C⁺⁺. Cette traduction est effectuée par l'unité de traduction
25 305.

A l'étape suivante E716, la méthode traduite est alors appelée (invoquée) par l'application 309 (celle-ci permet de mettre en œuvre les objets publiés dans la station serveur) sur l'objet identifié par l'URL extraite de la requête (E700).

30 L'étape suivante E718 est une étape de test dans laquelle on évalue le résultat de l'exécution de la méthode. Si l'exécution de la méthode

produit une erreur, une exception destinée à signaler l'erreur produite est encodée (étape E720).

Si le résultat produit est valide, on encode ce résultat (étape E722). L'encodage du résultat (ou de l'exception) consiste à traduire le résultat
5 obtenu du langage C++ en langage XML. Cette traduction est encore une fois assurée par l'unité de traduction 305.

Une fois le résultat ou l'exception encodée, le message (HTTP) de réponse ("message d'exécution de méthode") est généré par l'unité 304 d'identification d'objets et envoyé par le serveur HTTP 302, via le réseau, à
10 destination de la station cliente à l'origine de la requête POST.

La station cliente reçoit alors cette réponse et la traite comme exposé supra (fig. 6, E614).

En référence maintenant à la **figure 8**, on va décrire un ordinateur adapté à mettre en œuvre un procédé d'utilisation à distance d'un objet
15 informatique, en conformité avec l'invention, et/ou un procédé d'exécution d'une fonction sur un objet informatique, en conformité avec l'invention.

Un tel ordinateur peut, en effet, comporter à la fois des moyens adaptés à la mise en œuvre d'un procédé d'utilisation à distance d'un objet informatique, et des moyens adaptés à la mise en œuvre d'un procédé
20 d'exécution d'une fonction sur un objet informatique. C'est le cas notamment, lorsque l'ordinateur, constituant une station dans un réseau de communication, est client pour l'accès à des objets distants, et serveur pour offrir à des stations distantes l'accès à des objets qu'il héberge.

Dans ce mode de réalisation, les moyens adaptés à la mise en œuvre des procédés selon l'invention, sont essentiellement des composants
25 logiciels ou programmes.

Par conséquent, ces composants logiciels comportent une ou plusieurs séquences d'instructions dont l'exécution par ledit ordinateur permet la mise en œuvre des procédés selon l'invention.

30 Dans la figure 8, l'ordinateur 40 qui peut être typiquement un micro-ordinateur ou une station de travail, comporte de façon classique une

unité centrale (CPU) 70, reliée à une mémoire morte (ROM) 71 et à une mémoire vive (RAM) 72, ainsi qu'à un bus de données 82.

Le bus de données 82 permet la communication entre les différents sous-éléments de l'ordinateur 40, ou les éléments qui lui sont reliés.

- 5 Cependant, la communication entre les différents sous-éléments de l'ordinateur n'est pas limitée au bus 82. En particulier, l'unité centrale 70 est susceptible de communiquer des instructions à tout sous-élément de l'ordinateur 40 directement ou par l'intermédiaire d'un autre sous-élément de l'ordinateur 40.

10 L'ordinateur 40 comporte une interface de communication 81 reliée à un réseau de communication 1 tel que l'Internet. Cette interface de communication 81 comprend par exemple un modem de type connu de l'homme de l'art. Via l'interface de communication 81, l'ordinateur 40 peut permettre, par exemple, l'accès à des objets qu'il héberge ou permettre à un utilisateur connecté à l'ordinateur d'accéder à des objets distants.

15 L'ordinateur 40 comporte également de façon classique un moyen de stockage de données tel que, par exemple, un disque dur 76. Il peut également comporter un lecteur de disquettes 77, un lecteur de CD-ROM 78 et un lecteur de cartes de format dit PC-CARD 79.

20 Une disquette 7, un disque compact (CD) 8, une carte 9 de type PC-CARD – destinées à être lues respectivement par le lecteur de disquettes 77, le lecteur de CD-ROM 78 et le lecteur de cartes 79 – ainsi que le disque dur 76, peuvent être utilisés pour le stockage d'objets informatiques et des applications logicielles qui les mettent en œuvre ; ainsi que pour le stockage du code logiciel permettant la mise en œuvre du ou des procédés selon l'invention.

25 Selon un mode préféré de réalisation, le code exécutable du programme permettant de mettre en œuvre le ou les procédés selon l'invention, est mémorisé dans le disque dur 76 (programme 7600).

Selon une variante de réalisation, le code exécutable de ce programme est stocké dans la ROM 71 (programme 7100).

30 Selon une autre variante de réalisation, le code exécutable du programme peut être téléchargé à partir du réseau de communication 1 via l'interface de communication 81 pour être mémorisé sur le disque dur 76.

L'interface de communication 81 comporte, selon un mode de réalisation préféré, un logiciel de navigation Internet (*Web browser*).

Lors de l'exécution du programme, les variables créées et modifiées sont mémorisées dans des registres de la RAM 72 (variables 7200).

5 L'ordinateur 40 comporte en outre un écran 73 permettant l'affichage d'une interface graphique entre le programme selon l'invention et l'utilisateur, notamment pour afficher une interface utilisateur (GUI) associée à un objet informatique utilisé à distance, en conformité avec l'invention. L'utilisateur pourra alors formuler des requêtes à l'aide par exemple d'un
10 dispositif de pointage tel qu'une souris 75, ou bien à l'aide d'un clavier 74.

L'ordinateur 40 comporte en outre divers périphériques, tels qu'une imprimante 51 permettant par exemple d'imprimer des documents téléchargés, ou un télécopieur 52. Ces périphériques sont reliés à l'ordinateur via une carte d'entrée/sortie 80.

15 Bien entendu, de nombreuses modifications peuvent être apportées au modes de réalisation de l'invention décrits ci-dessus sans sortir du cadre de l'invention. En particulier, on peut prévoir d'associer à chacun des objets informatiques publiés dans une station serveur, plusieurs documents électroniques (documents XUL) décrivant chacun une interface utilisateur
20 différente de l'objet. Chacune des interfaces utilisateur de l'objet étant adaptée aux capacités du type de station cliente considéré. Par exemple, l'interface utilisateur de l'objet pourra être simplifiée si la station cliente est un téléphone portable.



ANNEXE

Annexe I - Exemple d'interface programmatique d'un objet informatique ("document XML").

5

```

<interface  name="ImageProcessor"
             href=http://tournesol:8080/web-obj/imageprocessor.xml>
  <functions>
10      <function name="buyImage" type="int">
          <arguments>
              <arg name="name" type="string"/>
          </arguments>
        </function>
15      <function name="negative" type="int">
          <arguments>
              <arg name="handle" type="int"/>
          </arguments>
        </function>
20      <function name="convertToBW" type="int">
          <arguments>
              <arg name="handle" type="int"/>
          </arguments>
        </function>
25      <function name="getImage" type="bool">
          <arguments>
              <arg name="handle" type="int"/>
          </arguments>
        </function>
30      </functions>
  </interface>

```

Annexe II - Exemple de "document XUL"

35

```
<box orient="vertical" width="400" height="350">
```

40

```
<!-- Fonctions Javascript -->
```

```
<html:script>
```

```

    function updatePrice() {
        account.text = "Account = " + pwstub.credit();
5    }

    function buyNewImage() {
        var name, short, h;
        name = imageToBuy.text;
10    h = pwstub.buyImage(name);
        imageList.add(name);
        handles.at_put(name, h);
        updatePrice();
15    }

    function getHandle() {
        return handles.at(imageList.selection());
20    }

    function applyFilter() {
        var handle;
        var filter, image;
        filter = effects.selection();
        image = imageList.selection();
25    handle = getHandle();
        Transcript.show(filter);
        Transcript.cr();
        if(filter == "Video Inverse") {
            handle = pwstub.negative(handle);
30    }
        if(filter == "Black and White") {
            handle = pwstub.convertToBW(handle);
        }
        handles.at_put(image, handle);
35    updatePrice();
    }

    function downloadImage() {
        var img;
40    Transcript.show("Downloading image");
        Transcript.cr();
        img = pwstub.getImage(getHandle());
        Transcript.show("Opening image");
        Transcript.cr();
45    IDX.openImage(img);
        Transcript.show("Image done");
        Transcript.cr();
        updatePrice();

```

```

    }

    function conn ctToServers() {
        var client;
5         RSA.setKeyLength(32);
        broker = Broker.newTo_port_path("tournesol", 8080,
            "/oml/obj/broker");
        server = RemoteServer.newTo_port_path("tournesol",
10        8080, "/oml/obj/imager");
        client = XULClient.new_name(broker, "Image Consumer");
        pwstub = client.subscribe_name_credit(server,
            "ImageProcessor", 30);
    }

15  <!-- Initialisation de l'interface utilisateur de l'objet -->

        var server, broker, pwstub;
        var imageToBuy, imageList, account, effects;
        var handles;

20

        imageToBuy = document.getElementById("imageToBuy");
        imageList = document.getElementById("imageList");
        account = document.getElementById("account");
        effects = document.getElementById("effects");

25

        handles = Dictionary.new();
        connectToServers();
        updatePrice();

30

        effects.add("Video Inverse");
        effects.add("Black and White");

    </html:script>

35  <!-- Définition des éléments de l'interface utilisateur de l'objet -->

        <box orient="vertical" width="390">

            <html:h1>Image Demo in XUL</html:h1>
40            <box orient="horizontal" height="80">
                <textfield width="320" id="imageToBuy" text=""/>
                <button value="Buy new"
oncommand="buyNewImage();" />
            </box>

45            <box orient="horizontal" height="170">
                <html:list width="200" height="130" id="imageList"/>
                <box orient="vertical" height="130">

```

```

                                <html:list height="100" id="effects"/>
                                <button value="Apply" height="30"
oncommand="applyFilter();" />
                                </box>
5      </box>
                                <button value="Download" oncommand="downloadImage();" />
                                <textfield id="account" text="" />
                                </box>
10     </box>

```

Annexe III - Exemple de contenu d'une "requête d'exécution de méthode"

```

15     <functions>
        <function name="buyImage:" />
        <arguments>
            <string >photo1.bmp</string>
20     </arguments>
    </functions>

```

Annexe IV : Exemple du contenu d'une "réponse d'exécution de méthode"

```

30     <wob>
        <int value="10" />
    </wob>

```

35

40

REVENDEICATIONS

- 5 1. Procédé d'utilisation à distance d'un objet informatique accessible via une station serveur connectée à un réseau de communication, à partir d'une station cliente connectée au réseau, le procédé étant caractérisé en ce qu'il comporte les étapes suivantes :
- génération et envoi (E500) d'un message de requête, dit
 - 10 "requête objet", vers la station serveur, ladite requête objet comportant une information d'identification d'un objet accessible via la station serveur ;
 - réception (E502) d'un message de réponse, dit "réponse objet", émis par la station serveur, ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des fonctions programmées,
 - 15 ladite interface permettant à un utilisateur d'utiliser ledit objet ;
 - lancement (E504, E506) de l'interface utilisateur sur la station cliente ;
 - exécution (E602) d'au moins une fonction associée à au moins un élément de l'interface utilisateur, en réponse à l'activation (E508, E600) dudit
 - 20 au moins un élément par un utilisateur ;
 - génération (E608, E610) et envoi (E612) à la station serveur d'un message de requête, dit "requête d'exécution de méthode", en réponse à l'exécution d'au moins une fonction programmée associée audit au moins un élément activé de l'interface utilisateur, ladite requête d'exécution de méthode
 - 25 comportant ladite information d'identification de l'objet et au moins une commande compréhensible par ledit objet.
2. Procédé selon la revendication 1, caractérisé en ce qu'il comporte en outre les étapes suivantes :
- 30 - réception (E614) d'un message de réponse, dit "réponse d'exécution de méthode", émis par la station serveur en réponse à la requête d'exécution de méthode, ladite réponse d'exécution de méthode contenant des

données indicatives du résultat de l'exécution de ladite au moins une commande compréhensible par ledit objet ;

- décodage (E620) des données contenues dans ladite réponse d'exécution de méthode et mise à jour (E622) si nécessaire de l'interface utilisateur.

3. Procédé d'exécution d'une fonction sur un objet informatique utilisable, via une station serveur connectée à un réseau de communication, par au moins une station cliente connectée audit réseau, caractérisé en ce qu'il comporte les étapes suivantes, mises en œuvre dans la station serveur :

- réception d'un message de requête, dit "requête objet", en provenance d'une station cliente, ladite requête objet comportant une information d'identification d'un objet informatique accessible via la station serveur ;
- génération et envoi à la station cliente d'un message de réponse, dit "réponse objet", ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des fonctions programmées, ladite interface permettant à un utilisateur d'utiliser ledit objet ;
- réception (E700) d'un message de requête, dit "requête d'exécution de méthode", en provenance de la station cliente, ladite requête d'exécution de méthode comportant ladite information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

4. Procédé selon la revendication 3, caractérisé en ce qu'il comporte en outre les étapes suivantes :

- exécution (E710-E716) de ladite au moins une commande, reçue de la station cliente, sur ledit objet informatique ;
- génération (E718-E724) et envoi (E726) à la station cliente, d'un message de réponse, dit "réponse d'exécution de méthode", ladite réponse d'exécution de méthode contenant des données indicatives du résultat de l'exécution de ladite au moins une commande sur ledit objet .

5. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que ladite commande compréhensible par ledit objet et contenue dans ladite requête d'exécution de méthode, consiste en la désignation d'une fonction de l'objet informatique considéré, et en la délivrance de valeurs données à des paramètres associés à la fonction désignée.

6. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que ladite information d'identification d'un objet comprend une adresse électronique indicative de l'emplacement de stockage dudit objet.

7. Procédé selon la revendication 6, caractérisé en ce que l'adresse électronique indicative de l'emplacement de stockage dudit objet, est une adresse de type URL.

8. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que lesdites informations de description d'une interface utilisateur sont des données exprimées dans un langage de communication informatique de type "langage de marquage".

9. Procédé selon la revendication 8, caractérisé en ce que ledit langage de communication est basé sur le langage XML.

10. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que, pour être accessible sur le réseau, ledit objet informatique est associé dans la station serveur à un document électronique (3071) contenant lesdites informations de description d'au moins une interface utilisateur et des fonctions programmées associées.

11. Procédé selon la revendication 10, caractérisé en ce que ledit document électronique (3071) est un document de type "document XML".

12. Procédé selon la revendication 10 ou 11, caractérisé en ce que ladite adresse électronique, indicative de l'emplacement de stockage dudit objet, est une adresse de type "URL" associée audit document électronique (3071).

5

13. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que ledit réseau de communication est un réseau du type Internet.

10

14. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que la station cliente et la station serveur communiquent en utilisant un protocole de communication de type "protocole de transfert hypertexte" (HTTP), et en ce que lesdits messages échangés entre la station serveur et la station cliente sont des messages HTTP.

15

15. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que lesdites fonctions programmées associées audit informations de description d'une interface utilisateur, sont implémentées dans le langage de programmation Javascript.

20

16. Dispositif (400) d'utilisation à distance d'un objet informatique accessible via une station serveur connectée à un réseau de communication, à partir d'une station cliente connectée au réseau, le dispositif étant caractérisé en ce qu'il comporte :

25

- des moyens (406, 402) pour générer et envoyer un message de requête, dit "requête objet", vers la station serveur, ladite requête objet comportant une information d'identification d'un objet accessible via la station serveur ;

30

- des moyens (402) pour recevoir un message de réponse, dit "réponse objet", émis par la station serveur, ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des

fonctions programmées, ladite interface permettant à un utilisateur d'utiliser ledit objet ;

- des moyens (404, 408, 410) pour lancer l'interface utilisateur sur la station cliente;

5 - des moyens (408, 410) pour exécuter au moins une fonction programmée associée à un élément de l'interface graphique, en réponse à l'activation dudit élément par un utilisateur ;

 - des moyens (402, 404, 406, 410) pour générer et envoyer à la station serveur un message de requête, dit "requête d'exécution de méthode",
10 ladite requête d'exécution de méthode comportant ladite information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

15 17. Dispositif selon la revendication 16, caractérisé en ce qu'il comporte en outre :

 - des moyens (402) pour recevoir, en réponse à la requête d'exécution de méthode, un message de réponse, dit "réponse d'exécution de méthode", émis par la station serveur, ladite réponse d'exécution de méthode contenant des données indicatives du résultat de l'exécution de ladite au moins
20 une commande compréhensible par ledit objet ;

 - des moyens (404) pour décoder les données contenues dans ladite réponse d'exécution de méthode ; et

 - des moyens (408, 410) pour mettre à jour si nécessaire l'interface utilisateur dudit objet.

25

 18. Dispositif selon la revendication 16 ou 17, caractérisé en ce qu'il comporte des moyens adaptés à la mise en œuvre d'un procédé d'utilisation à distance d'un objet informatique selon l'une quelconque des revendications 5 à 15.

30

 19. Dispositif (300) d'exécution d'une fonction sur un objet informatique utilisable, via une station serveur connectée à un réseau de

communication, par au moins une station cliente connectée audit réseau, caractérisé en ce qu'il comporte :

- des moyens (302) pour recevoir un message de requête, dit "requête objet", en provenance d'une station cliente, ladite requête objet comportant une information d'identification d'un objet informatique accessible via la station serveur ;
- des moyens (302, 304) pour générer et envoyer à la station cliente un message de réponse, dit "réponse objet", ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des fonctions programmées, ladite interface permettant à un utilisateur d'utiliser ledit objet préalablement identifié par ladite information d'identification ;
- des moyens (302) pour recevoir un message de requête, dit "requête d'exécution de méthode", en provenance de la station cliente, ladite requête d'exécution de méthode comportant ladite information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

20. Dispositif selon la revendication 19, caractérisé en ce qu'il comporte en outre :

- des moyens (304, 305) pour exécuter ladite au moins une commande, reçue de la station cliente, sur ledit objet informatique et obtenir un résultat ;
- des moyens (302, 304, 305) pour générer et envoyer, à la station cliente, un message de réponse, dit "réponse d'exécution de méthode", ladite réponse d'exécution de méthode contenant des données indicatives du résultat de l'exécution de ladite au moins une commande sur ledit objet .

21. Dispositif selon la revendication 19 ou 20, caractérisé en ce qu'il comporte des moyens adaptés à la mise en œuvre d'un procédé d'exécution d'une fonction sur un objet informatique, selon l'une quelconque des revendications 5 à 15.

22. Système informatique comportant un dispositif d'utilisation à distance d'un objet informatique, selon l'une quelconque des revendications 16 à 18, et/ou un dispositif d'exécution d'une fonction sur un objet informatique, selon l'une quelconque des revendications 19 à 21.

5

23. Dispositif de navigation sur l'Internet (navigateur Web) comportant un dispositif d'utilisation à distance d'un objet informatique, selon l'une quelconque des revendications 16 à 18.

10

24. Station cliente reliée à un réseau de communication, caractérisée en ce qu'elle comporte un dispositif d'utilisation à distance d'un objet informatique, selon l'une quelconque des revendications 16 à 18.

15

25. Station serveur reliée à un réseau de communication, caractérisée en ce qu'elle comporte un dispositif d'exécution d'une fonction sur un objet informatique, selon l'une quelconque des revendications 19 à 21.

20

26. Réseau de communication comportant au moins une station cliente selon la revendication 24, et au moins une station serveur selon la revendication 25.

REVENDEICATIONS

1. Procédé d'utilisation à distance d'un objet informatique accessible via une station serveur connectée à un réseau de communication, à partir d'une station cliente connectée au réseau, le procédé étant caractérisé en ce qu'il comporte les étapes suivantes :

- génération et envoi (E500) d'un message de requête, dit "requête objet", vers la station serveur, ladite requête objet comportant une information d'identification d'un objet accessible via la station serveur ;

- réception (E502) d'un message de réponse, dit "réponse objet", émis par la station serveur, ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des fonctions programmées, ladite interface permettant à un utilisateur d'utiliser ledit objet ;

- lancement (E504, E506) de l'interface utilisateur sur la station cliente ;

- exécution (E602) d'au moins une fonction associée à au moins un élément de l'interface utilisateur, en réponse à l'activation (E508, E600) dudit au moins un élément par un utilisateur ;

- génération (E608, E610) et envoi (E612) à la station serveur d'un message de requête, dit "requête d'exécution de méthode", en réponse à l'exécution d'au moins une fonction programmée associée audit au moins un élément activé de l'interface utilisateur, ladite requête d'exécution de méthode comportant ladite information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

2. Procédé selon la revendication 1, caractérisé en ce qu'il comporte en outre les étapes suivantes :

- réception (E614) d'un message de réponse, dit "réponse d'exécution de méthode", émis par la station serveur en réponse à la requête d'exécution de méthode, ladite réponse d'exécution de méthode contenant des

données indicatives du résultat de l'exécution de ladite au moins une commande compréhensible par ledit objet ;

- décodage (E620) des données contenues dans ladite réponse d'exécution de méthode et mise à jour (E622) si nécessaire de l'interface utilisateur.

3. Procédé selon la revendication 1 ou 2, caractérisé en ce que ladite commande compréhensible par ledit objet et contenue dans ladite requête d'exécution de méthode, consiste en la désignation d'une fonction de l'objet informatique considéré, et en la délivrance de valeurs données à des paramètres associés à la fonction désignée.

4. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que ladite information d'identification d'un objet comprend une adresse électronique indicative de l'emplacement de stockage dudit objet.

5. Procédé selon la revendication 4, caractérisé en ce que l'adresse électronique indicative de l'emplacement de stockage dudit objet, est une adresse de type URL.

6. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que lesdites informations de description d'une interface utilisateur sont des données exprimées dans un langage de communication informatique de type "langage de marquage".

7. Procédé selon la revendication 6, caractérisé en ce que ledit langage de communication est basé sur le langage XML.

8. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que, pour être accessible sur le réseau, ledit objet informatique est associé dans la station serveur à un document

électronique (3071) contenant lesdites informations de description d'au moins une interface utilisateur et des fonctions programmées associées.

9. Procédé selon la revendication 8, caractérisé en ce que ledit document électronique (3071) est un document de type "document XML".

10. Procédé selon la revendication 8 ou 9, caractérisé en ce que ladite adresse électronique, indicative de l'emplacement de stockage dudit objet, est une adresse de type "URL" associée audit document électronique (3071).

11. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que ledit réseau de communication est un réseau du type Internet.

12. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que la station cliente et la station serveur communiquent en utilisant un protocole de communication de type "protocole de transfert hypertexte" (HTTP), et en ce que lesdits messages échangés entre la station serveur et la station cliente sont des messages HTTP.

13. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que lesdites fonctions programmées associées audit informations de description d'une interface utilisateur, sont implémentées dans le langage de programmation Javascript.

14. Procédé d'exécution d'une fonction sur un objet informatique utilisable, via une station serveur connectée à un réseau de communication, par au moins une station cliente connectée audit réseau, caractérisé en ce qu'il comporte les étapes suivantes, mises en œuvre dans la station serveur :

- réception d'un message de requête, dit "requête objet", en provenance d'une station cliente, ladite requête objet comportant une

information d'identification d'un objet informatique accessible via la station serveur ;

- génération et envoi à la station cliente d'un message de réponse, dit "réponse objet", ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des fonctions programmées, ladite interface permettant à un utilisateur d'utiliser ledit objet ;

- réception (E700) d'un message de requête, dit "requête d'exécution de méthode", en provenance de la station cliente, ladite requête d'exécution de méthode comportant ladite information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

15. Procédé selon la revendication 14, caractérisé en ce qu'il comporte en outre les étapes suivantes :

- exécution (E710-E716) de ladite au moins une commande, reçue de la station cliente, sur ledit objet informatique ;

- génération (E718-E724) et envoi (E726) à la station cliente, d'un message de réponse, dit "réponse d'exécution de méthode", ladite réponse d'exécution de méthode contenant des données indicatives du résultat de l'exécution de ladite au moins une commande sur ledit objet .

16. Procédé selon la revendication 14 ou 15, caractérisé en ce que ladite commande compréhensible par ledit objet et contenue dans ladite requête d'exécution de méthode, consiste en la désignation d'une fonction de l'objet informatique considéré, et en la délivrance de valeurs données à des paramètres associés à la fonction désignée.

17. Procédé selon la revendication 14, 15 ou 16, caractérisé en ce que ladite information d'identification d'un objet comprend une adresse électronique indicative de l'emplacement de stockage dudit objet.

18. Procédé selon la revendication 17, caractérisé en ce que l'adresse électronique indicative de l'emplacement de stockage dudit objet, est une adresse de type URL.

19. Procédé selon l'une quelconque des revendications 14 à 18, caractérisé en ce que lesdites informations de description d'une interface utilisateur sont des données exprimées dans un langage de communication informatique de type "langage de marquage".

20. Procédé selon la revendication 19, caractérisé en ce que ledit langage de communication est basé sur le langage XML.

21. Procédé selon l'une quelconque des revendications 14 à 20, caractérisé en ce que, pour être accessible sur le réseau, ledit objet informatique est associé dans la station serveur à un document électronique (3071) contenant lesdites informations de description d'au moins une interface utilisateur et des fonctions programmées associées.

22. Procédé selon la revendication 21, caractérisé en ce que ledit document électronique (3071) est un document de type "document XML".

23. Procédé selon la revendication 21 ou 22, caractérisé en ce que ladite adresse électronique, indicative de l'emplacement de stockage dudit objet, est une adresse de type "URL" associée audit document électronique (3071).

24. Procédé selon l'une quelconque des revendications 14 à 23 précédentes, caractérisé en ce que ledit réseau de communication est un réseau du type Internet.

25. Procédé selon l'une quelconque des revendications 14 à 24, caractérisé en ce que la station cliente et la station serveur communiquent en

utilisant un protocole de communication de type "protocole de transfert hypertexte" (HTTP), et en ce que lesdits messages échangées entre la station serveur et la station cliente sont des messages HTTP.

26. Procédé selon l'une quelconque des revendications 14 à 25, caractérisé en ce que lesdites fonctions programmées associées audit informations de description d'une interface utilisateur, sont implémentées dans le langage de programmation Javascript.

27. Dispositif (400) d'utilisation à distance d'un objet informatique accessible via une station serveur connectée à un réseau de communication, à partir d'une station cliente connectée au réseau, le dispositif étant caractérisé en ce qu'il comporte :

- des moyens (406, 402) pour générer et envoyer un message de requête, dit "requête objet", vers la station serveur, ladite requête objet comportant une information d'identification d'un objet accessible via la station serveur ;

- des moyens (402) pour recevoir un message de réponse, dit "réponse objet", émis par la station serveur, ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des fonctions programmées, ladite interface permettant à un utilisateur d'utiliser ledit objet ;

- des moyens (404, 408, 410) pour lancer l'interface utilisateur sur la station cliente;

- des moyens (408, 410) pour exécuter au moins une fonction programmée associée à un élément de l'interface graphique, en réponse à l'activation dudit élément par un utilisateur ;

- des moyens (402, 404, 406, 410) pour générer et envoyer à la station serveur un message de requête, dit "requête d'exécution de méthode", ladite requête d'exécution de méthode comportant ladite information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

28. Dispositif selon la revendication 27, caractérisé en ce qu'il comporte en outre :

- des moyens (402) pour recevoir, en réponse à la requête d'exécution de méthode, un message de réponse, dit "réponse d'exécution de méthode", émis par la station serveur, ladite réponse d'exécution de méthode contenant des données indicatives du résultat de l'exécution de ladite au moins une commande compréhensible par ledit objet ;
- des moyens (404) pour décoder les données contenues dans ladite réponse d'exécution de méthode ; et
- des moyens (408, 410) pour mettre à jour si nécessaire l'interface utilisateur dudit objet.

29. Dispositif selon la revendication 27 ou 28, caractérisé en ce qu'il comporte des moyens adaptés à la mise en œuvre d'un procédé d'utilisation à distance d'un objet informatique selon l'une quelconque des revendications 3 à 13.

30. Dispositif (300) d'exécution d'une fonction sur un objet informatique utilisable, via une station serveur connectée à un réseau de communication, par au moins une station cliente connectée audit réseau, caractérisé en ce qu'il comporte :

- des moyens (302) pour recevoir un message de requête, dit "requête objet", en provenance d'une station cliente, ladite requête objet comportant une information d'identification d'un objet informatique accessible via la station serveur ;
- des moyens (302, 304) pour générer et envoyer à la station cliente un message de réponse, dit "réponse objet", ladite réponse objet comportant des informations de description d'une interface utilisateur associées à des fonctions programmées, ladite interface permettant à un utilisateur d'utiliser ledit objet préalablement identifié par ladite information d'identification ;

- des moyens (302) pour recevoir un message de requête, dit "requête d'exécution de méthode", en provenance de la station cliente, ladite requête d'exécution de méthode comportant ladite information d'identification de l'objet et au moins une commande compréhensible par ledit objet.

31. Dispositif selon la revendication 30, caractérisé en ce qu'il comporte en outre :

- des moyens (304, 305) pour exécuter ladite au moins une commande, reçue de la station cliente, sur ledit objet informatique et obtenir un résultat ;

- des moyens (302, 304, 305) pour générer et envoyer, à la station cliente, un message de réponse, dit "réponse d'exécution de méthode", ladite réponse d'exécution de méthode contenant des données indicatives du résultat de l'exécution de ladite au moins une commande sur ledit objet .

32. Dispositif selon la revendication 30 ou 31, caractérisé en ce qu'il comporte des moyens adaptés à la mise en œuvre d'un procédé d'exécution d'une fonction sur un objet informatique, selon l'une quelconque des revendications 16 à 26.

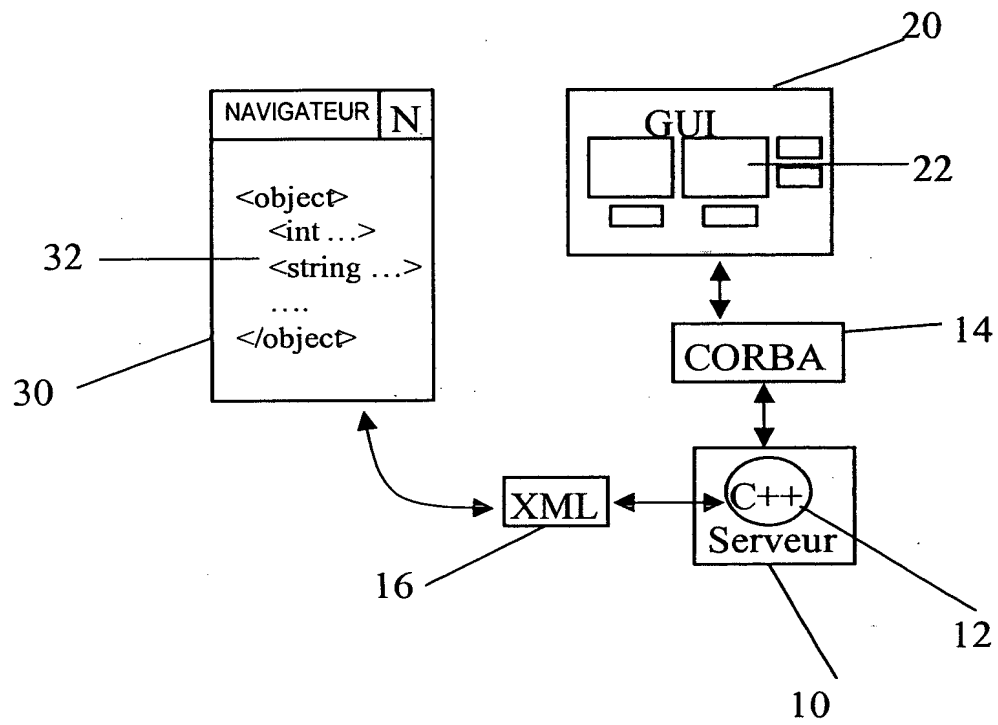
33. Système informatique comportant un dispositif d'utilisation à distance d'un objet informatique, selon l'une quelconque des revendications 27 à 29, et/ou un dispositif d'exécution d'une fonction sur un objet informatique, selon l'une quelconque des revendications 30 à 32.

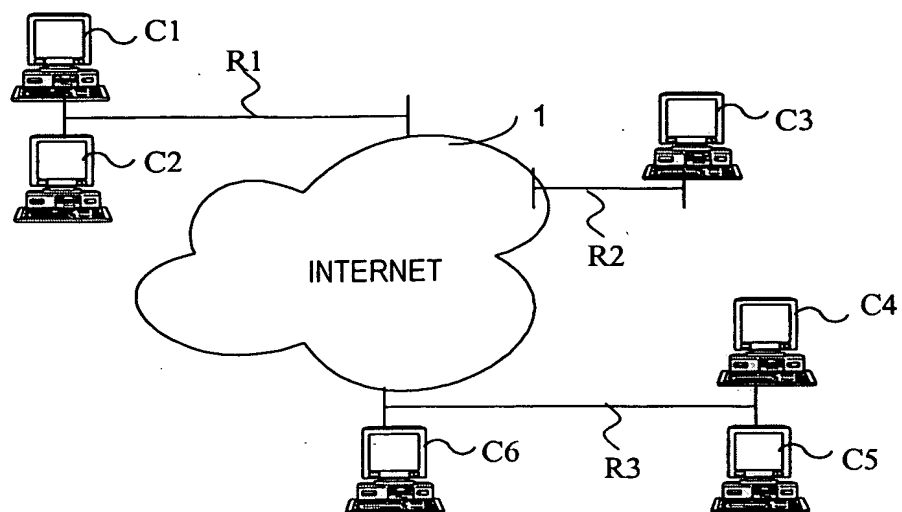
34. Dispositif de navigation sur l'Internet (navigateur Web) comportant un dispositif d'utilisation à distance d'un objet informatique, selon l'une quelconque des revendications 27 à 29.

35. Station cliente reliée à un réseau de communication, caractérisée en ce qu'elle comporte un dispositif d'utilisation à distance d'un objet informatique, selon l'une quelconque des revendications 27 à 29.

36. Station serveur reliée à un réseau de communication, caractérisée en ce qu'elle comporte un dispositif d'exécution d'une fonction sur un objet informatique, selon l'une quelconque des revendications 30 à 32.

37. Réseau de communication comportant au moins une station cliente selon la revendication 35, et au moins une station serveur selon la revendication 36.

**FIG. 1**

**FIG. 2**

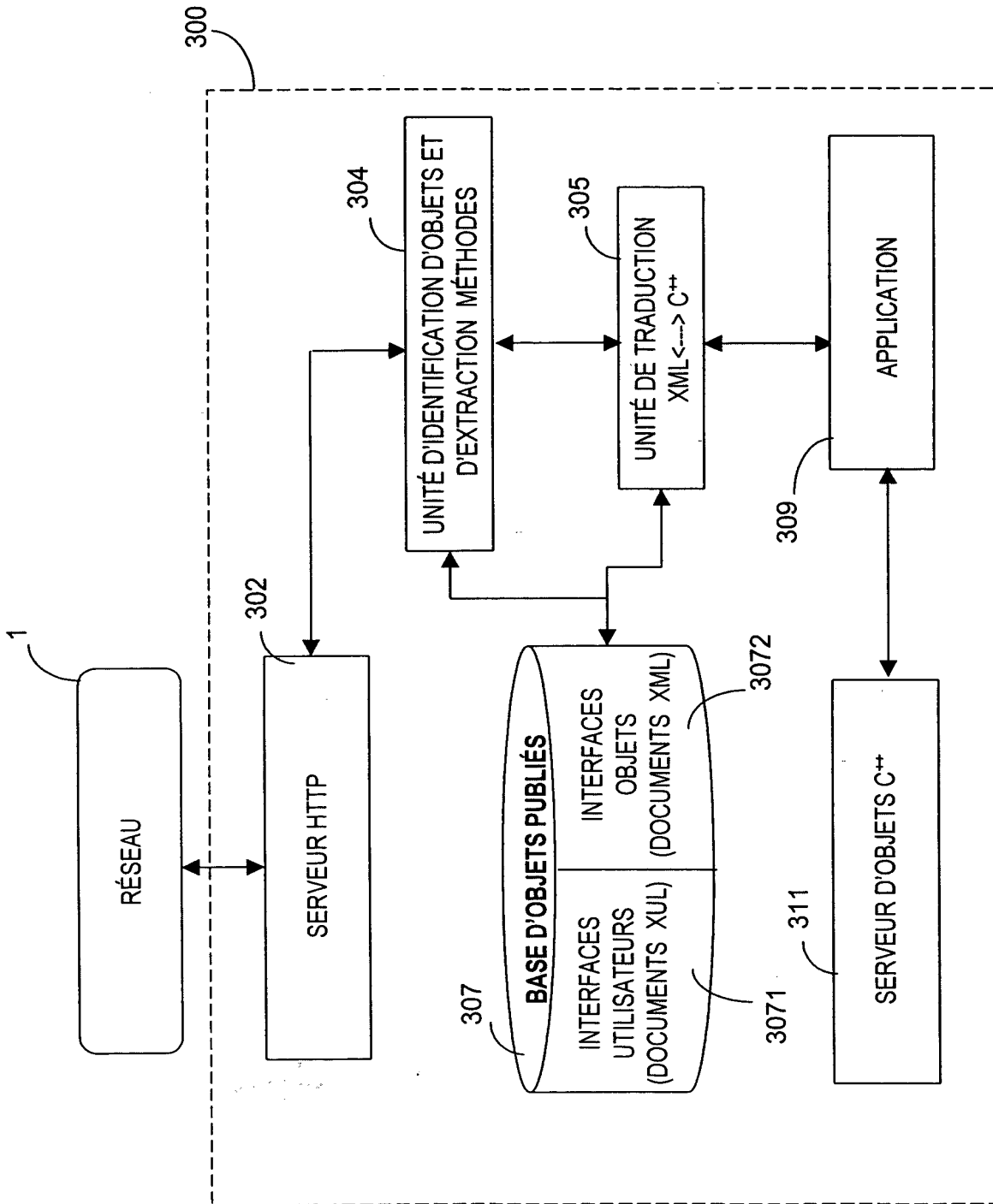


FIG. 3

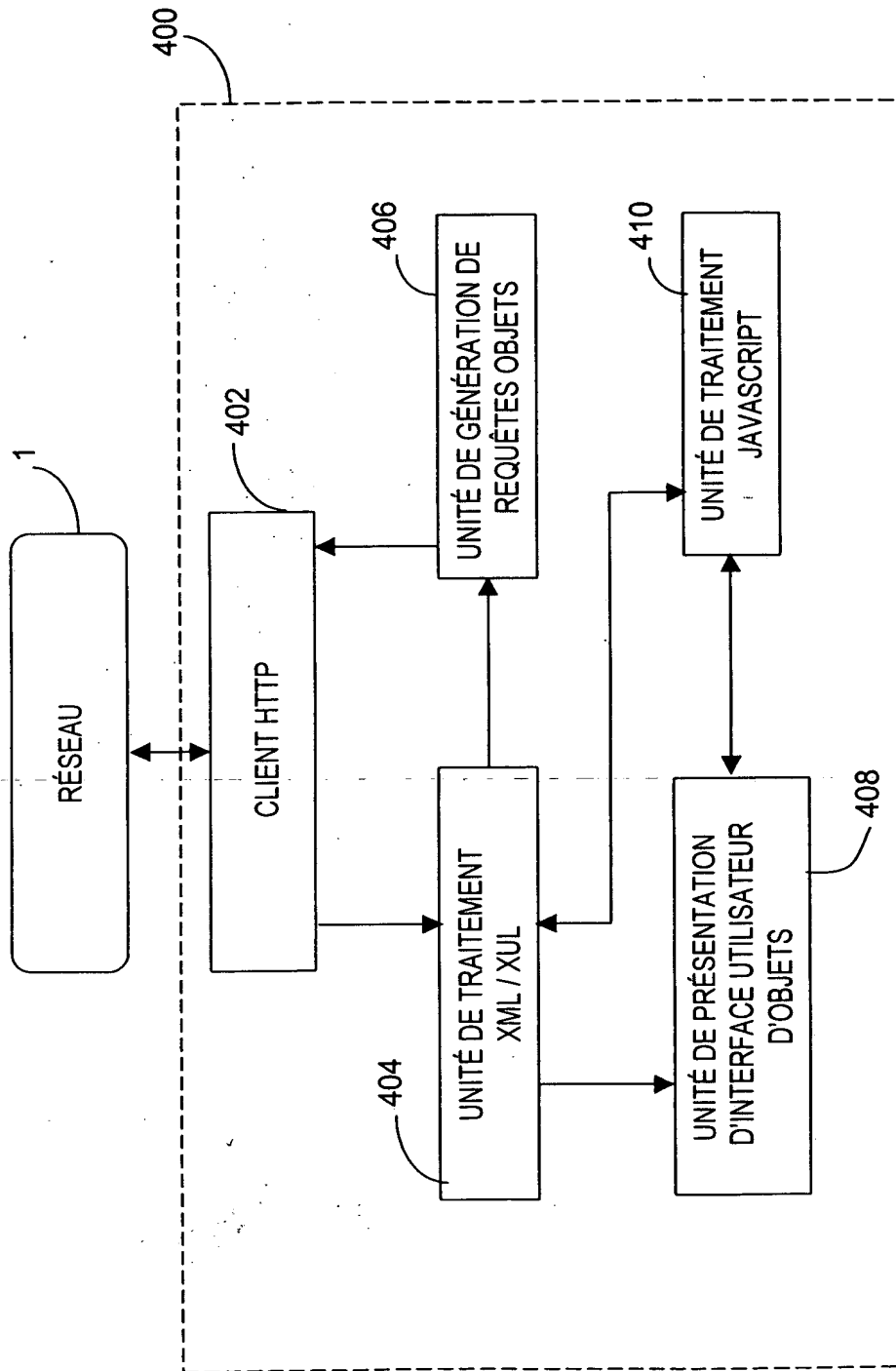
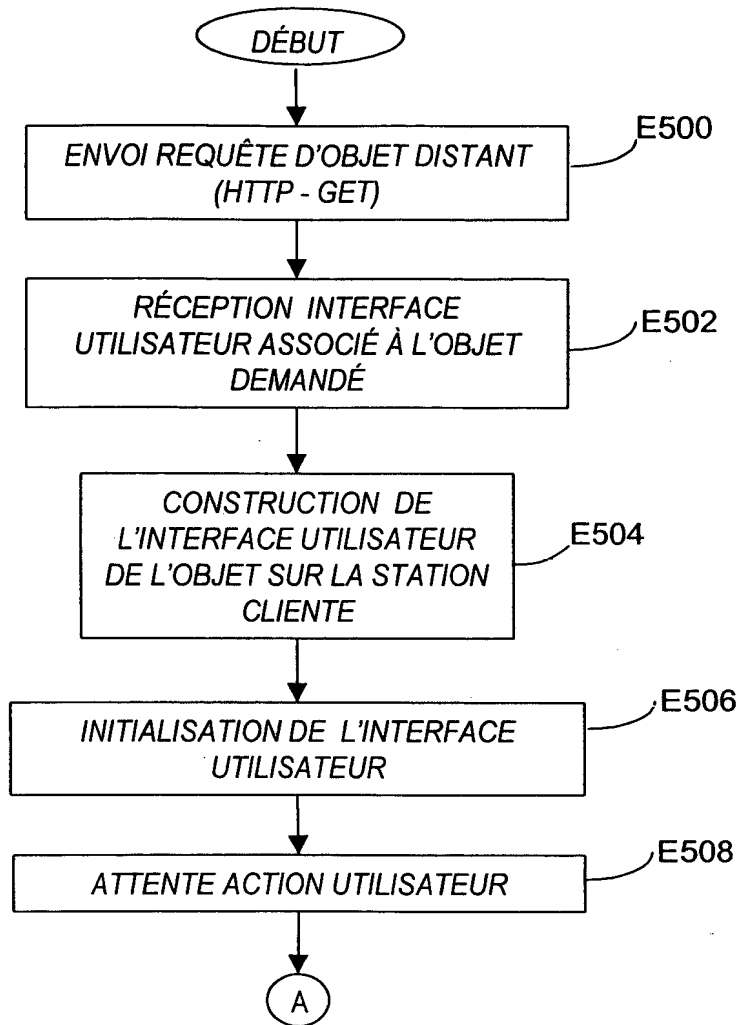


FIG. 4

**FIG. 5**

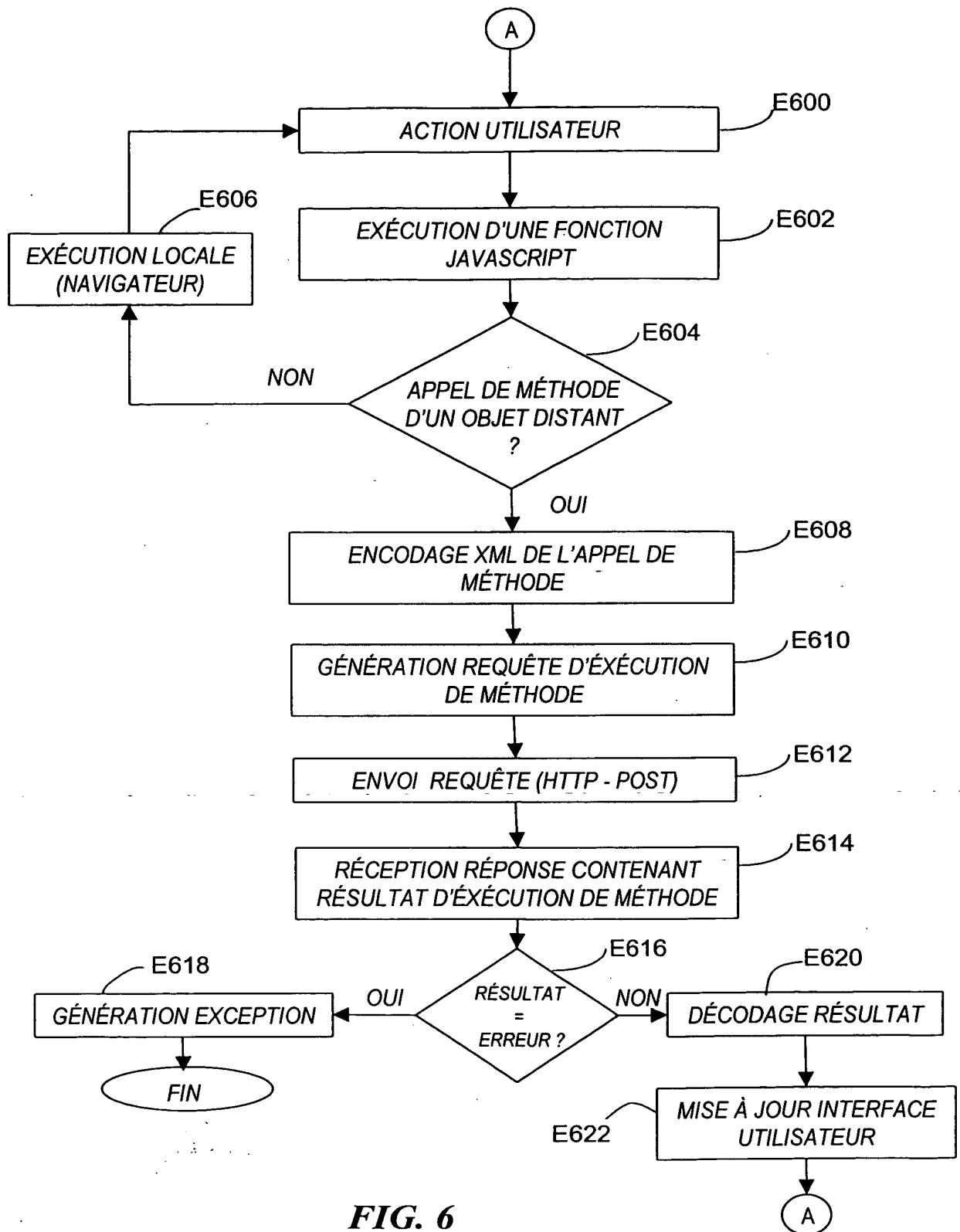


FIG. 6

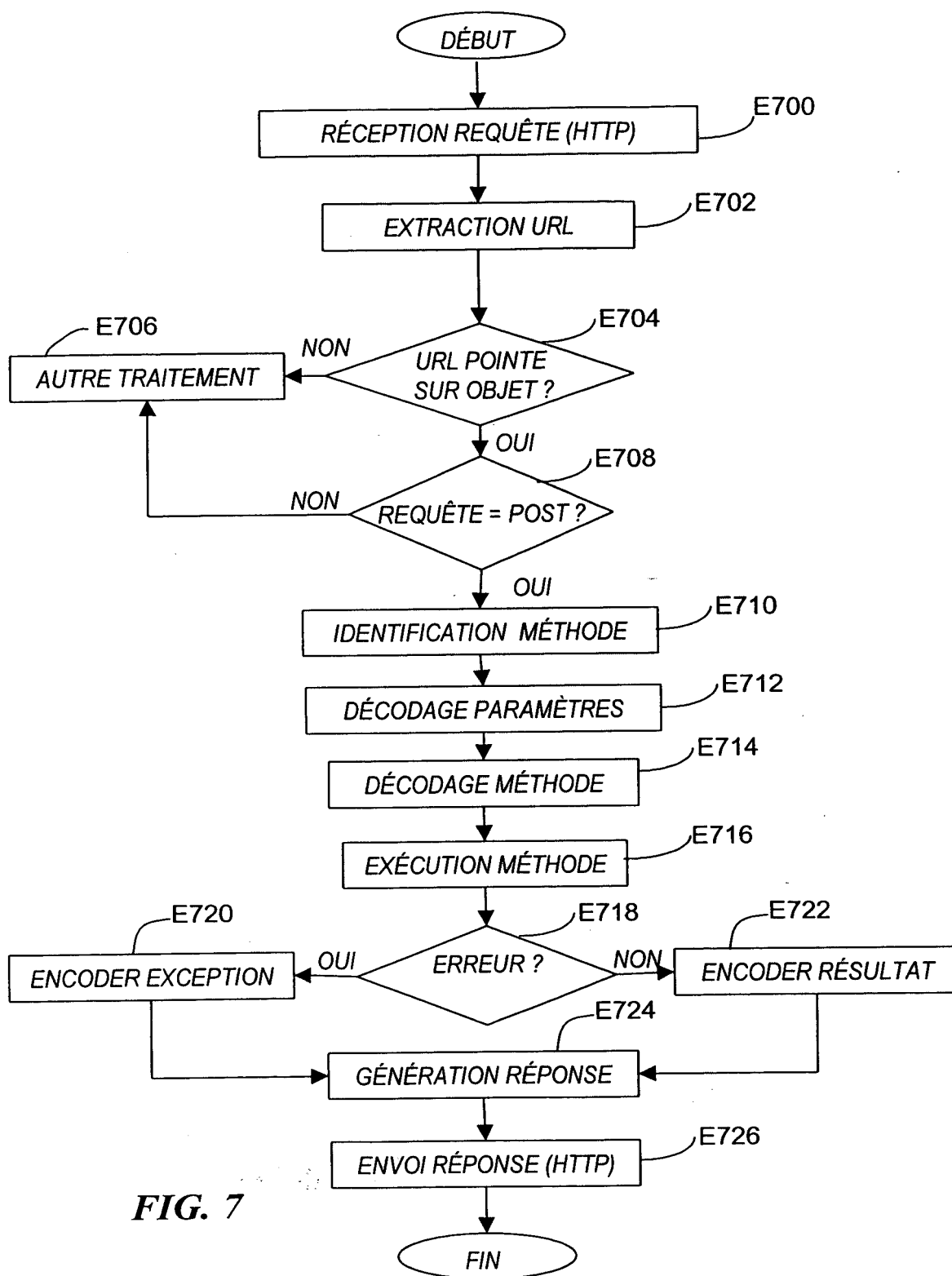


FIG. 7

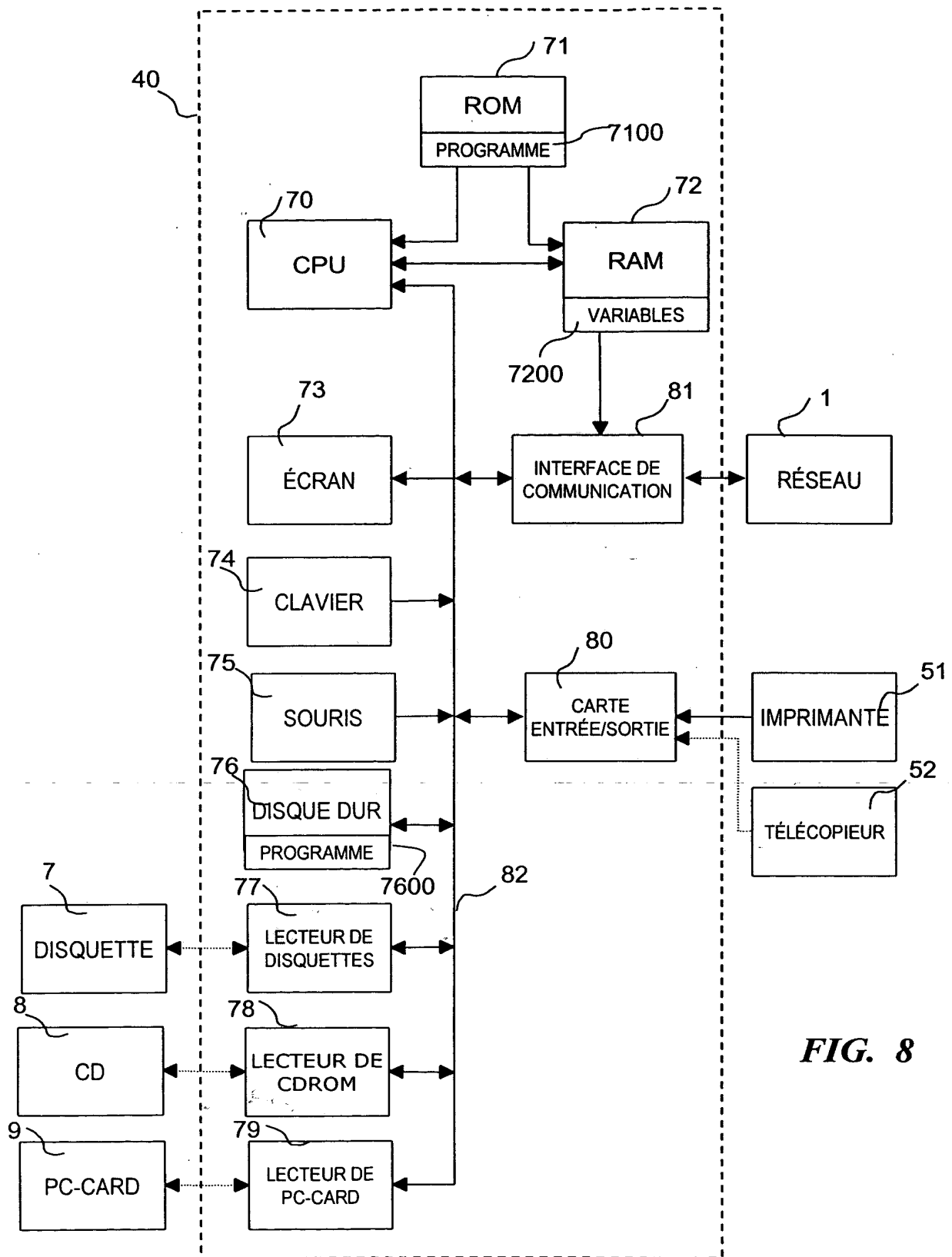


FIG. 8

THIS PAGE BLANK (USPTO,